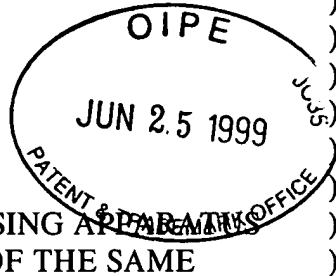


## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In the Patent Application of )  
)TOSHIKAZU KUROSE )  
)Serial No. 09/283,231 )  
)Filed: April 1, 1999 )  
)For: IMAGE PROCESSING APPARATUS )  
)  
AND METHOD OF THE SAME )ATTN:  
APPLICATION BRANCHCLAIM TO PRIORITY UNDER 35 U.S.C. 119Assistant Commissioner for Patents  
Washington, D.C. 20231

Sir:

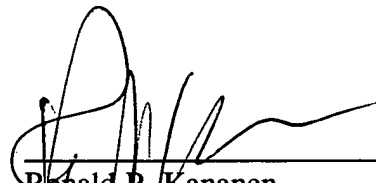
The benefit of the filing date of the following prior applications filed in the following foreign country is hereby requested and the right of priority provided under 35 U.S.C. 119 is hereby claimed:

Japanese Patent Appl. No. P10-091844, filed April 3, 1998

Japanese Patent Appl. No. P10-051795, filed February 26, 1999

In support of this claim, filed herewith is a certified copy of said original foreign applications.

Respectfully submitted,

  
\_\_\_\_\_  
Ronald P. Kananen  
Reg. No. 24,104

Dated: June 25, 1999

**RADER, FISHMAN & GRAUER P.L.L.C.**  
1233 20<sup>TH</sup> Street, NW  
Suite 501  
Washington, DC 20036  
202-955-3750-Phone  
202-955-3751 - Fax

## 日 本 国 特 許 庁

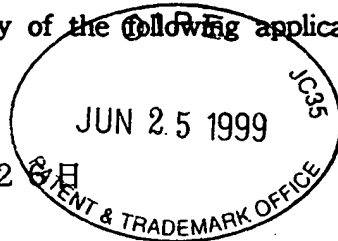
PATENT OFFICE  
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日  
Date of Application:

1999年 2月28日



出 願 番 号  
Application Number:

平成11年特許願第051795号

出 願 人  
Applicant (s):

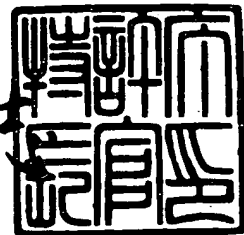
ソニー株式会社

CERTIFIED COPY OF  
PRIORITY DOCUMENT

1999年 3月26日

特許庁長官  
Commissioner,  
Patent Office

山 佐 建 志



【書類名】 特許願

【整理番号】 9900088312

【提出日】 平成11年 2月26日

【あて先】 特許庁長官殿

【国際特許分類】 H04N 15/00

【発明の名称】 画像処理装置およびその方法

【請求項の数】 40

【発明者】

    【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
                                内

    【氏名】 黒瀬 悦和

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

    【代表者】 出井 伸之

【代理人】

    【識別番号】 100094053

    【弁理士】

    【氏名又は名称】 佐藤 隆久

【先の出願に基づく優先権主張】

    【出願番号】 平成10年特許願第 91844号

    【出願日】 平成10年 4月 3日

【手数料の表示】

    【予納台帳番号】 014890

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

特平 1 1 - 0 5 1 7 9 5

【包括委任状番号】 9707389

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 画像処理装置およびその方法

【特許請求の範囲】

【請求項 1】

同時に処理を行うとする複数の画素毎にそれぞれ設けられ、入力した複数の第 1 の画素データを相互に並列に処理して複数の第 2 の画素データを生成する複数の画素処理回路と、

前記画素処理回路に入力する前記第 1 の画素データに基づいて、前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路の動作を停止する制御手段と

を有する画像処理装置。

【請求項 2】

前記画素処理回路は、クロック信号に基づいて動作し、

前記制御手段は、前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行うと判断した場合に、当該画素処理回路に前記クロック信号を供給し、

前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路への前記クロック信号の供給を停止する請求項 1 に記載の画像処理装置。

【請求項 3】

前記画素処理回路の各々は、パイプライン処理を行うように相互に直列に接続された複数の処理回路を有する

請求項 2 に記載の画像処理装置。

【請求項 4】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 3 に記載の画像処理装置。

【請求項 5】

前記画素処理回路は、画素の R（赤），G（緑），B（青）の出力を決定する画素データについての処理を行う

請求項 1 に記載の画像処理装置。

【請求項 6】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる画像処理装置において、

同時に処理を行なおうとする前記複数の画素データのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断する画素位置判断手段と

前記同時に処理を行おうとする前記複数の画素データを相互に並列に処理する複数の画素処理手段と、

前記画素位置判断手段の判断結果に基づいて、前記複数の画素処理手段のうち前記処理対象となっている単位図形の内側に位置する画素データを処理する前記画素処理手段以外の前記画素処理手段の動作を停止する制御手段と

を有する画像処理装置。

【請求項 7】

前記画素処理手段は、クロック信号に基づいて動作し、

前記制御手段は、前記処理対象となっている単位図形の内側に位置する画素の画素データを処理する前記画素処理手段にクロック信号を供給し、

前記処理対象となっている単位図形の内側に位置しない画素の画素データを処理する前記画素処理手段への前記クロック信号の供給を停止する

請求項 6 に記載の画像処理装置。

【請求項 8】

前記画素処理回路の各々は、パイプライン処理を行うように相互に直列に接続された複数の処理回路を有する

請求項 7 に記載の画像処理装置。

【請求項 9】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 8 に記載の画像処理装置。

【請求項 1 0】

前記画素位置判断手段は、前記画素処理手段で処理される画素データに、前記判断の結果を示す有効性指示データを付加し、

前記制御手段は、前記有効性指示データに基づいて、前記画素処理手段の動作を停止するか否かを判断する

請求項 6 に記載の画像処理装置。

【請求項 1 1】

同時に処理を行うとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データと対応する複数の第 2 の画素データとを、各画素毎に設定された混合比データが示す混合比で混合して複数の第 3 の画素データを生成する複数の画素処理手段と、

前記混合比データに基づいて、前記画素処理手段による前記混合を行うか否かを判断し、前記混合を行わないと判断した場合に、当該画素処理手段の動作を停止する制御手段と

を有する画像処理装置。

【請求項 1 2】

前記画素処理回路は、クロック信号に基づいて動作し、

前記制御手段は、前記混合を行うと判断した場合に、当該画素処理手段に前記クロック信号を供給し、

前記混合を行わないと判断した場合に、当該画素処理手段への前記クロック信号の供給を停止する

請求項 1 1 に記載の画像処理装置。

【請求項 1 3】

前記画素処理回路の各々は、パイプライン処理を行うように相互に直列に接続された複数の処理回路を有する

請求項 1 2 に記載の画像処理装置。

【請求項 1 4】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 1 3 に記載の画像処理装置。

【請求項 1 5】

前記第 2 の画素データを記憶する記憶手段

をさらに有し、

前記制御手段は、前記混合を行わないと判断した場合に、前記第 1 の画素データによって前記記憶手段に記憶されている前記第 2 の画素データを書き換えるように制御し、

前記混合を行うと判断した場合に、前記第 3 の画素データによって前記記憶手段に記憶されている前記第 2 の画素データを書き換えるように制御する

請求項 1 1 に記載の画像処理装置。

【請求項 1 6】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、

同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データと対応する複数の第 2 の画素データとを、各画素毎に予め設定された混合比データが示す混合比で混合して複数の第 3 の画素データを生成する複数の画素処理手段と、



前記同時に処理を行なおうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記混合比データに基づいて前記混合を行わないと判断した場合に、当該画素についての画素処理を行う画素処理手段の動作を停止する制御手段と

を有する画像処理装置。

【請求項 17】

記憶手段と、

同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第1の画素データから複数の第2の画素データを生成する複数の画素処理手段と、

複数の前記第1の画素データの複数の第1の奥行きデータと、前記複数の第1の奥行きデータに対応して前記記憶手段に記憶されている複数の第3の画素データの複数の第2の奥行きデータとをそれぞれ比較する比較手段と、

前記比較の結果に基づいて、前記記憶手段に記憶されている前記第2の奥行きデータに対応した第3の画素データを前記第2の画素データによって書き換えるか否かを判断し、書き換えないと判断した場合に、対応する画素処理手段の動作を停止する制御手段と

を有する画像処理装置。

【請求項 18】

前記画素処理回路は、クロック信号に基づいて動作し、

前記制御手段は、前記記憶手段に記憶されている前記第3の画素データを前記第2の画素データによって書き換えると判断した場合に、対応する前記画素処理手段に前記クロック信号を供給し、

前記記憶手段に記憶されている前記第3の画素データを前記第2の画素データによって書き換えないと判断した場合に、対応する画素処理手段への前記クロック信号の供給を停止する

請求項 17に記載の画像処理装置。

【請求項 19】

前記画素処理回路の各々は、パイプライン処理を行うように相互に直列に接続

された複数の処理回路を有する

請求項 1 8 に記載の画像処理装置。

【請求項 2 0】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 1 9 に記載の画像処理装置。

【請求項 2 1】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、

記憶手段と、

同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データから複数の第 2 の画素データを生成する複数の画素処理手段と、

前記複数の第 1 の画素データの複数の第 1 の奥行きデータと、前記複数の第 1 の奥行きデータに対応して前記記憶手段に記憶されている複数の第 3 の画素データの複数の第 2 の奥行きデータとを比較する比較手段と、

前記同時に処理を行なおうとする前記複数の画素のデータのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記比較結果に基づいて前記記憶手段に記憶されている前記第 2 の奥行きデータに対応した第 3 の画素データを前記第 2 の画素データによって書き換えるか否かを判断し、当該判断の結果、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記書き換えを行わないと判断した場合に、対応する画素処理手段の動作を停止する制御手段と

を有する画像処理装置。

【請求項 22】

同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データから複数の第 2 の画素データを生成する複数の画素処理手段を用いて画像処理を行う画像処理方法において、

前記画素処理回路が入力する前記第 1 の画素データに基づいて、前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行うか否かを判断し、

前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路の動作を停止する

画像処理方法。

【請求項 23】

前記画素処理回路がクロック信号に基づいて動作し、

前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行うと判断した場合に、当該画素処理回路にクロック信号を供給し、

前記第 2 の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路へのクロック信号の供給を停止する

請求項 22 に記載の画像処理方法。

【請求項 24】

前記画素処理回路の各々は、直列に接続された複数の処理回路でパイプライン処理を行う

請求項 23 に記載の画像処理方法。

【請求項 25】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 24 に記載の画像処理方法。

【請求項 26】

前記画素処理は、画素の R（赤）、G（緑）、B（青）の出力を決定する画素データについての処理を行う

請求項 22 に記載の画像処理方法。

【請求項 27】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる画像処理方法において、

同時に処理を行なおうとする前記複数の画素データのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、

前記同時に処理を行なおうとする前記複数の画素のデータを複数の画素処理手段において相互に並列に処理し、

前記判断の結果に基づいて、前記複数の画素処理手段のうち前記処理対象となっている単位図形の内側に位置する画素の画素データを処理する前記画素処理手段以外の前記画素処理手段の動作を停止する

画像処理方法。

【請求項 28】

前記画素処理手段がクロック信号に基づいて動作し、

前記処理対象となっている単位図形の内側に位置する画素の画素データを処理する前記画素処理手段にクロック信号を供給し、

前記処理対象となっている単位図形の内側に位置しない画素の画素データを処理する前記画素処理手段への前記クロック信号の供給を停止する

請求項 27 に記載の画像処理方法。

【請求項 29】

前記画素処理回路の各々は、直列に接続された複数の処理回路でパイプライン処理を行う

請求項 28 に記載の画像処理方法。

【請求項 30】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレ

ジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 29 に記載の画像処理方法。

【請求項 31】

同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第 1 の画素データと複数の第 2 の画素データとを、各画素毎に予め設定された混合比データが示す混合比で混合して複数の第 3 の画素データを生成し、

前記混合比データに基づいて、前記画素処理手段による前記混合を行うか否かを判断し、

前記混合を行わないと判断した場合に、対応する画素処理手段の動作を停止する

画像処理方法。

【請求項 32】

前記画素処理回路がクロック信号に基づいて動作し、

前記混合を行うと判断した場合に、対応する画素処理手段に前記クロック信号を供給し、

前記混合を行わないと判断した場合に、対応する画素処理手段への前記クロック信号の供給を停止する

請求項 31 に記載の画像処理方法。

【請求項 33】

前記画素処理回路の各々は、直列に接続された複数の処理回路でパイプライン処理を行う

請求項 32 に記載の画像処理方法。

【請求項 34】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 33 に記載の画像処理方法。

【請求項 35】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、

同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第 1 の画素データと複数の第 2 の画素データとを、各画素毎に予め設定された混合比データが示す混合比で混合して複数の第 3 の画素データを生成し、

前記同時に処理を行なおうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記混合比データに基づいて前記混合を行わないと判断した場合に、対応する前記画素処理手段の動作を停止する

画像処理方法。

【請求項 36】

同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第 1 の画素データから複数の第 2 の画素データを生成し、

前記複数の第 1 の画素データの複数の第 1 の奥行きデータと、前記複数の第 1 の奥行きデータに対応して記憶手段に記憶されている複数の第 3 の画素データの複数の第 2 の奥行きデータとをそれぞれ比較し、

前記比較の結果に基づいて、前記記憶手段に記憶されている前記第 2 の奥行きデータに対応した第 3 の画素データを前記第 2 の画素データによって書き換えるか否かを判断し、書き換えないと判断した場合に、対応する画素処理手段の動作を停止する

画像処理方法。

【請求項 3 7】

前記画素処理回路がクロック信号に基づいて動作し、

前記記憶手段に記憶されている前記第 3 の画素データを前記第 2 の画素データによって書き換えると判断した場合に、対応する前記第 2 の画素処理手段に前記クロック信号を供給し、

前記記憶手段に記憶されている前記第 3 の画素データを前記第 2 の画素データによって書き換えないと判断した場合に、対応する前記第 2 の画素処理手段への前記クロック信号の供給を停止する

請求項 3 6 に記載の画像処理方法。

【請求項 3 8】

前記画素処理回路の各々は、直列に接続された複数の処理回路でパイプライン処理を行う

請求項 3 7 に記載の画像処理方法。

【請求項 3 9】

前記画素処理回路内の直列に接続された複数の処理回路は、それぞれフラグ記憶部を有し、前記複数の処理回路の前記フラグ記憶部が直列接続されてシフトレジスタを構成し、当該シフトレジスタにより前記パイプライン処理および前記クロック信号の供給の制御を行う

請求項 3 8 に記載の画像処理方法。

【請求項 4 0】

表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、

同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第 1 の画素データから複数の第 2 の画素データを生成し、

前記複数の前記第 1 の画素データの前記複数の第 1 の奥行きデータと、前記複数の第 1 の奥行きデータに対応して記憶手段に記憶されている複数の第 3 の画素

データの複数の第 2 の奥行きデータとをそれぞれ比較し、

前記同時に処理を行おうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記比較の結果に基づいて、前記記憶手段に記憶されている前記第 2 の奥行きデータに対応した第 3 の画素データを前記第 2 の画素データによって書き換えるか否かを判断し、当該判断の結果、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記書き換えを行わないと判断した場合に、対応する画素処理手段の動作を停止する

画像処理方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、低消費電力化を図れる画像処理装置およびその方法に関する。

【0 0 0 2】

【従来の技術】

種々の C A D (Computer Aided Design) システムや、アミューズメント装置などにおいて、コンピュータグラフィックスがしばしば用いられている。特に、近年の画像処理技術の進展に伴い、3次元コンピュータグラフィックスを用いたシステムが急速に普及している。

このような 3 次元コンピュータグラフィックスでは、各画素（ピクセル）に対応する色を決定するときに、各画素の色の値を計算し、この計算した色の値を、当該画素に対応するディスプレイバッファ（フレームバッファ）のアドレスに書き込むレンダリング (Rendering) 処理を行う。

レンダリング処理の手法の一つに、ポリゴン (Polygon) レンダリングがある。この手法では、立体モデルを三角形の単位図形（ポリゴン）の組み合わせで表現し、このポリゴンを単位として処理を行い、描画することで、表示画面の色を決定する。

【0 0 0 3】

ポリゴンレンダリングでは、物理座標系における三角形の各頂点についての、



座標  $(x, y, z)$  と、色データ  $(R, G, B, \alpha)$  と、張り合わせのイメージパターンを示すテクスチャデータの同次座標  $(s, t)$  および同次項  $q$  の値とを入力とし、これらの値を三角形の内部で補間する処理が行われる。

ここで、同次項  $q$  は、簡単にいうと、拡大縮小率のようなもので、実際のテクスチャバッファの  $UV$  座標系における座標、すなわち、テクスチャ座標データ  $(u, v)$  は、同次座標  $(s, t)$  を同次項  $q$  で除算した「 $s/q$ 」および「 $t/q$ 」に、それぞれテクスチャサイズ  $U\ S\ I\ Z\ E$  および  $V\ S\ I\ Z\ E$  を乗じたものとなる。

このような 3 次元コンピュータグラフィックシステムでは、例えば、ディスプレイバッファ（フレームバッファ）に描画を行う際に、画素毎に、テクスチャ座標データ  $(u, v)$  を用いてテクスチャバッファからテクスチャデータを読み出し、この読み出したテクスチャデータを、立体モデルの表面に三角形を単位として張り付けるテクスチャマッピング処理を行う。

なお、立体モデルでのテクスチャマッピング処理では、各画素毎に、張り付けを行なうテクスチャデータが示す画像の拡大縮小率が変化する。

#### 【0004】

ところで、このような 3 次元コンピュータグラフィックシステムでは、例えば、所定の矩形内の 8 画素についての処理を並行して（同時に）行う場合がある。

また、前述したような三角形を単位図形としたポリゴンレンダリングでは、張り付けを行うテクスチャデータの縮小率などは、三角形を単位として決定される。

従って、並行して処理を行った 8 画素分の演算結果のうち、対象となる三角形の外部に位置する画素についての演算結果は無効になる。

具体的には、図 12 に示すように、三角形 30 について所定の演算を行って縮小率を決定し、当該縮小率に応じたテクスチャデータを用いてテクスチャマッピング処理を行っている場合を考える。

ここで、矩形 31, 32, 33 は、それぞれ並行して処理される 8 ( $2 \times 4$ ) 画素が配置された領域であり、ポリゴンレンダリング処理において、各矩形内に属する 8 画素については同じテクスチャデータが用いられる。

図 12 に示す場合には、矩形 32 に属する 8 画素は全て三角形 30 内に位置するため、8 画素の演算結果は全て有効「1」である。これに対して、矩形 31、33 にそれぞれ属する 8 画素は、3 画素は三角形 30 内に位置するが、5 画素は三角形 30 の外に位置する。従って、8 画素の演算結果のうち、3 画素の演算結果は有効であるが、5 画素の演算結果は無効となる。

従来では、矩形内に位置する 8 画素の全てについて、ポリゴンレンダリング処理を無条件に行っていた。

#### 【0005】

##### 【発明が解決しようとする課題】

しかしながら、上述したように、三角形を単位図形としたポリゴンレンダリング処理を行なう場合に、矩形内に位置する複数の画素の全てについての処理を、対象となっている三角形の内部に位置するか否かとは無関係に実行すると、膨大な数の無効な演算を行うことになり、消費電力に大きな影響を及ぼす。

また、3次元コンピュータグラフィックシステムでは、上述した理由の他にも、種々の要因で不要な演算を行うことがある。

また、近年、3次元コンピュータグラフィックシステムの動作クロック周波数は非常に高くなっているため、消費電力の低下が大きな課題になっている。

#### 【0006】

本発明は上述した従来技術の問題点に鑑みてなされ、消費電力の大幅な低下を図れる画像処理装置およびその方法を提供することを目的とする。

#### 【0007】

##### 【課題を解決するための手段】

上述した従来技術の問題点を解決し、上述した目的を達成するために、本発明の第1の観点の画像処理装置は、同時に処理を行うとする複数の画素毎にそれぞれ設けられ、入力した複数の第1の画素データを相互に並列に処理して複数の第2の画素データを生成する複数の画素処理回路と、前記画素処理回路に入力する前記第1の画素データに基づいて、前記第2の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路の動作を停止する制御手段とを有する。

## 【 0 0 0 8 】

また、本発明の第 2 の観点の画像処理装置は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる画像処理装置であって、同時に処理を行なおうとする前記複数の画素データのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断する画素位置判断手段と、前記同時に処理を行なおうとする前記複数の画素データを相互に並列に処理する複数の画素処理手段と、前記画素位置判断手段の判断結果に基づいて、前記複数の画素処理手段のうち前記処理対象となっている単位図形の内側に位置する画素データを処理する前記画素処理手段以外の前記画素処理手段の動作を停止する制御手段とを有する。

## 【 0 0 0 9 】

また、本発明の第 3 の観点の画像処理装置は、同時に処理を行うとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データと対応する複数の第 2 の画素データとを、各画素毎に設定された混合比データが示す混合比で混合して複数の第 3 の画素データを生成する複数の画素処理手段と、前記混合比データに基づいて、前記画素処理手段による前記混合を行うか否かを判断し、前記混合を行わないと判断した場合に、当該画素処理手段の動作を停止する制御手段とを有する。

## 【 0 0 1 0 】

また、本発明の第 4 の観点の画像処理装置は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データと対応する複数の第 2 の画素データとを、各画素毎に予め設定された混合比データが示す混

合比で混合して複数の第 3 の画素データを生成する複数の画素処理手段と、前記同時に処理を行なおうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記混合比データに基づいて前記混合を行わないと判断した場合に、当該画素についての画素処理を行う画素処理手段の動作を停止する制御手段とを有する。

## 【 0 0 1 1 】

また、本発明の第 5 の観点の画像処理装置は、記憶手段と、同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データから複数の第 2 の画素データを生成する複数の画素処理手段と、複数の前記第 1 の画素データの複数の第 1 の奥行きデータと、前記複数の第 1 の奥行きデータに対応して前記記憶手段に記憶されている複数の第 3 の画素データの複数の第 2 の奥行きデータとをそれぞれ比較する比較手段と、前記比較の結果に基づいて、前記記憶手段に記憶されている前記第 2 の奥行きデータに対応した第 3 の画素データを前記第 2 の画素データによって書き換えるか否かを判断し、書き換えないと判断した場合に、対応する画素処理手段の動作を停止する制御手段とを有する。

## 【 0 0 1 2 】

また、本発明の第 6 の観点の画像処理装置は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、記憶手段と、同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第 1 の画素データから複数の第 2 の画素データを生成する複数の画素処理手段と、前記複数の第 1 の画素データの複数の第 1 の奥行きデータと、前記複数の第 1 の奥行きデータに対応して前記記憶手段に記憶されている複数の第 3 の画素データの複数の第 2 の奥行きデータとを比較する比較手段と、前記同時に処理を行なおうとする前記複数の画素のデータのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記比較結果に基づいて前記記憶手段に記憶されてい

る前記第2の奥行きデータに対応した第3の画素データを前記第2の画素データによって書き換えるか否かを判断し、当該判断の結果、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又はかつ前記書き換えを行わないと判断した場合に、対応する画素処理手段の動作を停止する制御手段とを有する。

#### 【0013】

また、本発明の第1の観点の画像処理方法は、同時に処理を行おうとする複数の画素毎にそれぞれ設けられ、複数の第1の画素データから複数の第2の画素データを生成する複数の画素処理手段を用いて画像処理を行う画像処理方法であって、前記画素処理回路が入力する前記第1の画素データに基づいて、前記第2の画素データを生成するための画素処理を前記画素処理回路が行うか否かを判断し、前記第2の画素データを生成するための画素処理を前記画素処理回路が行わないと判断した場合に、当該画素処理回路の動作を停止する。

#### 【0014】

また、本発明の第2の観点の画像処理方法は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる画像処理方法であって、同時に処理を行なおうとする前記複数の画素データのそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記同時に処理を行おうとする前記複数の画素のデータを複数の画素処理手段において相互に並列に処理し、前記判断の結果に基づいて、前記複数の画素処理手段のうち前記処理対象となっている単位図形の内側に位置する画素の画素データを処理する前記画素処理手段以外の前記画素処理手段の動作を停止する。

#### 【0015】

また、本発明の第3の観点の画像処理方法は、同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第1の画素データと複数の第2の画素データとを、各画素毎に予め設定された混合比データが

示す混合比で混合して複数の第3の画素データを生成し、前記混合比データに基づいて、前記画素処理手段による前記混合を行うか否かを判断し、前記混合を行わないと判断した場合に、対応する画素処理手段の動作を停止する。

## 【0016】

また、本発明の第4の観点の画像処理方法は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第1の画素データと複数の第2の画素データとを、各画素毎に予め設定された混合比データが示す混合比で混合して複数の第3の画素データを生成し、前記同時に処理を行なおうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記混合比データに基づいて前記混合を行わないと判断した場合に、対応する前記画素処理手段の動作を停止する。

## 【0017】

また、本発明の第5の観点の画像処理方法は、同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第1の画素データから複数の第2の画素データを生成し、前記複数の第1の画素データの複数の第1の奥行きデータと、前記複数の第1の奥行きデータに対応して記憶手段に記憶されている複数の第3の画素データの複数の第2の奥行きデータとをそれぞれ比較し、前記比較の結果に基づいて、前記記憶手段に記憶されている前記第2の奥行きデータに対応した第3の画素データを前記第2の画素データによって書き換えるか否かを判断し、書き換えないと判断した場合に、対応する画素処理手段の動作を停止する。

## 【0018】

また、本発明の第6の観点の画像処理方法は、表示手段に表示する画像を所定形状の単位図形を組み合わせて表現し、同一の前記単位図形内に位置する複数の

画素の画素データを同一の処理条件に基づいて処理し、同時に処理を行う複数の画素の画素データのうち処理対象となっている前記単位図形の内側に位置する画素の画素データの処理結果を有効なものとして用いる場合に、同時に処理を行おうとする複数の画素毎にそれぞれ設けられた複数の画素処理手段により、複数の第1の画素データから複数の第2の画素データを生成し、前記複数の前記第1の画素データの前記複数の第1の奥行きデータと、前記複数の第1の奥行きデータに対応して前記記憶手段に記憶されている複数の第3の画素データの複数の第2の奥行きデータとをそれぞれ比較し、前記同時に処理を行おうとする前記複数の画素のそれぞれについて、対応する画素が前記単位図形の内側に位置するか否かを判断し、前記比較の結果に基づいて、前記記憶手段に記憶されている前記第2の奥行きデータに対応した第3の画素データを前記第2の画素データによって書き換えるか否かを判断し、当該判断の結果、前記対応する画素が前記単位図形の内側に位置しないと判断した場合、又は前記書き換えを行わないと判断した場合に、対応する画素処理手段の動作を停止する。

【0019】

#### 【発明の実施の形態】

以下、本実施形態においては、家庭用ゲーム機などに適用される、任意の3次元物体モデルに対する所望の3次元画像をCRT (Cathode Ray Tube)などのディスプレイ上に高速に表示する3次元コンピュータグラフィックシステムについて説明する。

#### 第1実施形態

図1は、本実施形態の3次元コンピュータグラフィックシステム1のシステム構成図である。

3次元コンピュータグラフィックシステム1は、立体モデルを単位図形である三角形（ポリゴン）の張り合わせとして表現し、このポリゴンを描画することで表示画面の各画素の色を決定し、ディスプレイに表示するポリゴンレンダリング処理を行うシステムである。

また、3次元コンピュータグラフィックシステム1では、平面上の位置を表現する(x, y)座標の他に、奥行きを表すz座標を用いて3次元物体を表し、こ

の  $(x, y, z)$  の 3 つの座標で 3 次元空間の任意の一点を特定する。

#### 【 0 0 2 0 】

図 1 に示すように、3 次元コンピュータグラフィックシステム 1 では、メインメモリ 2、I/O インタフェース回路 3、メインプロセッサ 4 およびレンダリング回路 5 が、メインバス 6 を介して接続されている。

以下、各構成要素の機能について説明する。

メインプロセッサ 4 は、例えば、ゲームの進行状況などに応じて、メインメモリ 2 から必要なグラフィックデータを読み出し、このグラフィックデータに対してクリッピング (Clipping) 処理、ライティング (Lighting) 処理およびジオメトリ (Geometry) 処理などを行い、ポリゴンレンダリングデータを生成する。メインプロセッサ 4 は、ポリゴンレンダリングデータ S 4 を、メインバス 6 を介してレンダリング回路 5 に出力する。

I/O インタフェース回路 3 は、必要に応じて、外部からポリゴンレンダリングデータを入力し、これをメインバス 6 を介してレンダリング回路 5 に出力する。

#### 【 0 0 2 1 】

ここで、ポリゴンレンダリングデータは、ポリゴンの各 3 頂点の  $(x, y, z, R, G, B, \alpha, s, t, q)$  のデータを含んでいる。

$(x, y, z)$  データは、ポリゴンの頂点の 3 次元座標を示し、 $(R, G, B)$  データは、それぞれ当該 3 次元座標における赤、緑、青の輝度値を示している。

$\alpha$  データは、これから描画する画素と、ディスプレイバッファ 2 1 に既に記憶されている画素との  $R, G, B$  データのブレンド (混合) 係数を示している。

$(s, t, q)$  データのうち、 $(s, t)$  は、対応するテクスチャの同次座標を示しており、 $q$  は同次項を示している。ここで、「 $s/q$ 」および「 $t/q$ 」に、それぞれテクスチャサイズ  $U S I Z E$  および  $V S I Z E$  を乗じてテクスチャ座標データ  $(u, v)$  が得られる。テクスチャバッファ 2 0 に記憶されたテクスチャデータへのアクセスは、テクスチャ座標データ  $(u, v)$  を用いて行われる。



すなわち、ポリゴンレンダリングデータは、三角形の各頂点の物理座標値と、それぞれの頂点の色とテクスチャデータの同次座標および同次項を示している。

#### 【0022】

以下、レンダリング回路5について詳細に説明する。

図1に示すように、レンダリング回路5は、DDA (Digital Differential Analyzer) セットアップ回路10、トライアングルDDA回路11、テクスチャエンジン回路12、メモリI/F回路13、CRTコントローラ回路14、RAMDAC回路15、DRAM16およびSRAM17を有する。

DRAM16は、テクスチャバッファ20、ディスプレイバッファ21、zバッファ22およびテクスチャCLUTバッファ23として機能する。

#### 【0023】

##### DDAセットアップ回路10

DDAセットアップ回路10は、後段のトライアングルDDA回路11において物理座標系上の三角形の各頂点の値を線形補間して三角形の内部の各画素の色と深さ情報を求めるのに先立ち、ポリゴンレンダリングデータS4が示す(z, R, G, B,  $\alpha$ , s, t, q) データについて、三角形の辺と水平方向の差分を求めるセットアップ演算を行う。

このセットアップ演算は、具体的には、開始点の値と終点の値と、開始点と終点との距離を用いて、単位長さ移動した場合における、求めようとしている値の変分を算出する。

#### 【0024】

また、DDAセットアップ回路10は、同時に処理を行う8画素のそれぞれについて、処理対象となる三角形の内部に位置するか否かを示す1ビットの有効指示データvalを決定する。具体的には、有効指示データvalは、三角形の内部に位置する画素について「1」とし、三角形の外部に位置する画素について「0」とする。

DDAセットアップ回路10は、算出した変分データS10と、各画素の有効指示データvalとをトライアングルDDA回路11に出力する。

【0025】

トライアングルDDA回路11

トライアングルDDA回路11は、DDAセットアップ回路10から入力した変分データS10を用いて、三角形内部の各画素の線形補間された(z, R, G, B,  $\alpha$ , s, t, q)データを算出する。

トライアングルDDA回路11は、各画素の(x, y)データと、当該(x, y)座標の画素についての(z, R, G, B,  $\alpha$ , s, t, q, val)データとを、DDAデータ(補間データ)S11としてテクスチャエンジン回路12に出力する。

本実施形態では、トライアングルDDA回路11は、並行して処理を行う矩形内に位置する8画素分のDDAデータS11を単位としてテクスチャエンジン回路12に出力する。

【0026】

ここで、DDAデータS11の(z, R, G, B,  $\alpha$ , s, t, q, val)データは、図2に示すように、161ビットのデータである。

具体的には、R, G, B,  $\alpha$ データがそれぞれ8ビットであり、z, s, t, qデータがそれぞれ32ビットであり、valデータが1ビットである。

なお、以下、並行して処理を行う8画素についての(z, R, G, B,  $\alpha$ , s, t, q, val)データのうち、valデータをvalデータS220<sub>1</sub>～S220<sub>8</sub>とし、(z, R, G, B,  $\alpha$ , s, t, q)データを被演算データS221<sub>1</sub>～S221<sub>8</sub>とする。

すなわち、トライアングルDDA回路11は、8画素分の(x, y)データと、valデータS220<sub>1</sub>～S220<sub>8</sub>と、被演算データS221<sub>1</sub>～S221<sub>8</sub>からなるDDAデータS11をテクスチャエンジン回路12に出力する。

【0027】

テクスチャエンジン回路12およびメモリI/F回路13

テクスチャエンジン回路12による、DDAデータS11を用いた、「s/q」および「t/q」の算出処理、テクスチャ座標データ(u, v)の算出処理、および、テクスチャバッファ20からの(R, G, B,  $\alpha$ )データの読み出し処

理と、メモリ I/F 回路 13 による z 比較処理および混合処理とを、図 3 に示す演算ブロック 200, 201, 202, 204, 205 でパイプライン方式で順に実行する。

ここで、演算ブロック 200, 201, 202, 204, 205 は、それぞれ 8 個の演算サブブロックを内蔵しており、8 画素分の演算処理を並行して行う。

ここで、テクスチャエンジン回路 12 が演算ブロック 200, 201, 202 を内蔵し、メモリ I/F 回路 13 が演算ブロック 204, 205 を内蔵している。

#### 【0028】

##### 〔演算ブロック 200〕

演算ブロック 200 は、DDA データ S11 に含まれる (s, t, q) データを用いて、s データを q データで除算する演算と、t データを q データで除算する演算とを行う。

演算ブロック 200 は、図 3 に示すように、8 個の演算サブブロック 200<sub>1</sub> ~ 200<sub>8</sub> を内蔵する。

ここで、演算サブブロック 200<sub>1</sub> は、被演算データ S221<sub>1</sub> および val データ S220<sub>1</sub> を入力し、val データ S220<sub>1</sub> が「1」、すなわち有効であることを示す場合には、「s/q」および「t/q」を算出し、その算出結果を除算結果 S200<sub>1</sub> として演算ブロック 201 の演算サブブロック 201<sub>1</sub> に出力する。

#### 【0029】

また、演算サブブロック 200<sub>1</sub> は、val データ S220<sub>1</sub> が「0」、すなわち無効であることを示す場合には、演算は行わず、除算結果 S200<sub>1</sub> を出力しないか、あるいは、所定の仮値を示す除算結果 S200<sub>1</sub> を演算ブロック 201 の演算サブブロック 201<sub>1</sub> に出力する。

また、演算サブブロック 200<sub>1</sub> は、val データ S220<sub>1</sub> を後段の演算サブブロック 201<sub>1</sub> に出力する。

なお、演算サブブロック 200<sub>2</sub> ~ 200<sub>8</sub> も、それぞれ対応する画素について、演算サブブロック 200<sub>1</sub> と同じ演算を行い、それぞれ除算結果 S200<sub>2</sub>

～S200<sub>8</sub> および val データ S220<sub>2</sub> ～S220<sub>8</sub> を後段の演算ブロック 201 の演算サブブロック 201<sub>2</sub> ～201<sub>8</sub> にそれぞれ出力する。

#### 【0030】

図4は、演算サブブロック 200<sub>1</sub> の内部構成図である。

なお、図3に示す、全ての演算サブブロックは、基本的に、図4に示す構成をしている。

図4に示すように、演算サブブロック 200<sub>1</sub> は、クロックイネーブラ 210<sub>1</sub>、データ用フリップフロップ 222、プロセッサエレメント 223 およびフラグ用フリップフロップ 224 を有する。

クロックイネーブラ 210<sub>1</sub> は、システムクロック信号 S225 を基準としたタイミングで val データ S220<sub>1</sub> を入力し、val データ S220<sub>1</sub> のレベルを検出する。そして、クロックイネーブラ 210<sub>1</sub> は、val データ S220<sub>1</sub> が、「1」である場合には、例えば、クロック信号 S210<sub>1</sub> にパルス発生させ、「0」である場合には、クロック信号 S210<sub>1</sub> にパルス発生させない。

#### 【0031】

データ用フリップフロップ 222 は、クロック信号 S210<sub>1</sub> のパルスを検出すると、被演算データ S221<sub>1</sub> を取り込み、プロセッサエレメント 223 に出力する。

プロセッサエレメント 223 は、入力した被演算データ S221<sub>1</sub> を用いて前述した除算を行い、除算結果 S200<sub>1</sub> を演算サブブロック 201<sub>1</sub> のデータ用フリップフロップ 222 に出力する。

フラグ用フリップフロップ 224 は、システムクロック信号 S225 を基準としたタイミングで、val データ S220<sub>1</sub> を取り込み、後段の演算ブロック 201 の演算サブブロック 201<sub>1</sub> のフラグ用フリップフロップ 224 に出力する。

なお、システムクロック信号 S225 は、図3に示す全ての演算サブブロック 200<sub>1</sub> ～200<sub>8</sub>，201<sub>1</sub> ～201<sub>8</sub>，202<sub>1</sub> ～202<sub>8</sub>，204<sub>1</sub> ～204<sub>8</sub> のクロックイネーブラおよびフラグ用フリップフロップ 224 に供給される。

すなわち、演算サブブロック $200_1 \sim 200_8$ ， $201_1 \sim 201_8$ ， $202_1 \sim 202_8$ ， $204_1 \sim 204_8$ における処理は同期して行われ、同一の演算ブロックに内蔵された8個の演算サブブロックは並行して処理を行う。

【0032】

〔演算ブロック201〕

演算ブロック201は、演算サブブロック $201_1 \sim 201_8$ を有し、演算ブロック200から入力した除算結果 $S200_1 \sim S200_8$ が示す「 $s/q$ 」および「 $t/q$ 」に、それぞれテクスチャサイズ $U\ S\ I\ Z\ E$ および $V\ S\ I\ Z\ E$ を乗じて、テクスチャ座標データ（ $u$ ， $v$ ）を生成する。

演算サブブロック $201_1 \sim 201_8$ は、それぞれクロックイネーブラ $211_1 \sim 211_8$ により $v\ a\ l$ データ $S220_1 \sim S220_8$ のレベル検出を行った結果、当該レベルが「1」の場合にのみ演算を行い、それぞれ演算結果であるテクスチャ座標データ $S201_1 \sim S201_8$ を、演算ブロック202の演算サブブロック $202_1 \sim 202_8$ に出力する。

【0033】

〔演算ブロック202〕

演算ブロック202は、演算サブブロック $202_1 \sim 202_8$ を有し、メモリI/F回路13を介して、SRAM17あるいはDRAM16に、演算ブロック201で生成したテクスチャ座標データ（ $u$ ， $v$ ）を含む読み出し要求を出力し、メモリI/F回路13を介して、SRAM17あるいはテクスチャバッファ20に記憶されているテクスチャデータを読み出すことで、（ $u$ ， $v$ ）データに対応したテクスチャアドレスに記憶された（ $R$ ， $G$ ， $B$ ， $\alpha$ ）データ $S17$ を得る。

なお、テクスチャバッファ20には、MIPMAP（複数解像度テクスチャ）などの複数の縮小率に対応したテクスチャデータが記憶されている。ここで、何れの縮小率のテクスチャデータを用いるかは、所定のアルゴリズムを用いて、前記三角形を単位として決定される。

また、SRAM17には、テクスチャバッファ20に記憶されているテクスチャデータのコピーが記憶されている。

演算サブブロック  $202_1 \sim 202_8$  は、それぞれクロックイネーブラ  $212_1 \sim 212_8$  により val データ  $S220_1 \sim S220_8$  のレベル検出を行った結果、当該レベルが「1」の場合にのみ読み出し処理を行い、それぞれ読み出した (R, G, B,  $\alpha$ ) データ  $S17$  を、(R, G, B,  $\alpha$ ) データ  $S202_1 \sim S202_8$  として、それぞれ演算ブロック 203 の演算サブブロック  $203_1 \sim 203_8$  に出力する。

#### 【0034】

なお、テクスチャエンジン回路 12 は、フルカラー方式の場合には、テクスチャバッファ 20 から読み出した (R, G, B,  $\alpha$ ) データを直接用いる。一方、テクスチャエンジン回路 12 は、インデックスカラー方式の場合には、予め作成したカラールックアップテーブル (CLUT) をテクスチャ CLUT バッファ 23 から読み出して、内蔵する SRAM に転送および記憶し、このカラールックアップテーブルを用いて、テクスチャバッファ 20 から読み出したカラーインデックスに対応する (R, G, B) データを得る。

#### 【0035】

##### 〔演算ブロック 203〕

演算ブロック 203 は、演算サブブロック  $203_1 \sim 203_8$  を有し、演算ブロック 202 から入力したテクスチャデータである (R, G, B,  $\alpha$ ) データ  $S202_1 \sim S202_8$  と、トライアングル DDA 回路 11 からの DDA データ  $S11$  に含まれる (R, G, B) データとを、(R, G, B,  $\alpha$ ) データ  $S202_1 \sim S202_8$  に含まれる  $\alpha$  データ (テクスチャ  $\alpha$ ) が示す割合で混合し、(R, G, B) 混合データを生成する。

そして、演算ブロック 203 は、生成された (R, G, B) 混合データと、対応する DDA データ  $S11$  に含まれる  $\alpha$  データとを含む (R, G, B,  $\alpha$ ) データ  $S203_1 \sim S203_8$  を、演算ブロック 204 に出力する。

演算サブブロック  $203_1 \sim 203_8$  は、それぞれクロックイネーブラ  $213_1 \sim 213_8$  により val データ  $S220_1 \sim S220_8$  のレベル検出を行った結果、当該レベルが「1」の場合にのみ上記混合および (R, G, B,  $\alpha$ ) データ  $S203_1 \sim S203_8$  の出力を行う。

## 【0036】

## 〔演算ブロック204〕

演算ブロック204は、演算サブブロック204<sub>1</sub>～204<sub>8</sub>を有し、入力した(R, G, B,  $\alpha$ )データS203<sub>1</sub>～S203<sub>8</sub>について、zバッファ22に記憶されたzデータの内容を用いて、z比較を行い、(R, G, B,  $\alpha$ )データS203<sub>1</sub>～S203<sub>8</sub>によって描画する画像が、前回、ディスプレイバッファ21に描画した値よりも手前(視点側)に位置する場合には、zバッファ22を更新すると共に、(R, G, B,  $\alpha$ )データS203<sub>1</sub>～S203<sub>8</sub>を、(R, G, B,  $\alpha$ )データS204<sub>1</sub>～S204<sub>8</sub>として、それぞれ演算ブロック205の演算サブブロック205<sub>1</sub>～205<sub>8</sub>に出力する。

演算サブブロック204<sub>1</sub>～204<sub>8</sub>は、それぞれクロックイネーブラ214<sub>1</sub>～214<sub>8</sub>によりvalデータS220<sub>1</sub>～S220<sub>8</sub>のレベル検出を行った結果、当該レベルが「1」の場合にのみ上述したz比較および(R, G, B,  $\alpha$ )データS204<sub>1</sub>～S204<sub>8</sub>の出力を行なう。

## 【0037】

## 〔演算ブロック205〕

演算ブロック205は、演算サブブロック205<sub>1</sub>～205<sub>8</sub>を有し、入力した(R, G, B,  $\alpha$ )データS204<sub>1</sub>～S204<sub>8</sub>と、既にディスプレイバッファ21に記憶されている(R, G, B)データとを、それぞれ(R, G, B,  $\alpha$ )データS204<sub>1</sub>～S204<sub>8</sub>に含まれる $\alpha$ データが示す混合値で混合し、混合後の(R, G, B)データS205<sub>1</sub>～S205<sub>8</sub>をディスプレイバッファ21に書き込む(打ち込む)。

なお、メモリI/F回路13によるDRAM16に対してのアクセスは、16画素について同時に行なわれる。

演算サブブロック205<sub>1</sub>～205<sub>8</sub>は、それぞれクロックイネーブラ215<sub>1</sub>～215<sub>8</sub>によりvalデータS220<sub>1</sub>～S220<sub>8</sub>のレベル検出を行った結果、当該レベルが「1」の場合にのみ上述した混合処理およびディスプレイバッファ21への書き込み処理を行う。

## 【0038】

CRTコントローラ回路 14

CRTコントローラ回路 14 は、与えられた水平および垂直同期信号に同期して、図示しない CRT に表示するアドレスを発生し、ディスプレイバッファ 21 から表示データを読み出す要求をメモリ I/F 回路 13 に出力する。この要求に応じて、メモリ I/F 回路 13 は、ディスプレイバッファ 21 から一定の固まりで表示データを読み出す。CRTコントローラ回路 14 は、ディスプレイバッファ 21 から読み出した表示データを記憶する FIFO (First In First Out) 回路を内蔵し、一定の時間間隔で、RAMDAC 回路 15 に、RGB のインデックス値を出力する。

## 【0039】

RAMDAC 回路 15

RAMDAC 回路 15 は、各インデックス値に対応する R, G, B データを記憶しており、CRTコントローラ回路 14 から入力した RGB のインデックス値に対応するデジタル形式の R, G, B データを、D/A コンバータに転送し、アナログ形式の R, G, B データを生成する。RAMDAC 回路 15 は、この生成された R, G, B データを CRT に出力する。

## 【0040】

以下、3次元コンピュータグラフィックシステム 1 の全体動作について説明する。

ポリゴンレンダリングデータ S4 が、メインバス 6 を介してメインプロセッサ 4 から DDA セットアップ回路 10 に出力され、DDA セットアップ回路 10 において、三角形の辺と水平方向の差分などを示す変分データ S10 が生成される。

この変分データ S10 は、トライアングル DDA 回路 11 に出力され、トライアングル DDA 回路 11 において、三角形内部の各画素における線形補間された  $(z, R, G, B, \alpha, s, t, q)$  データが算出される。そして、この算出された  $(z, R, G, B, \alpha, s, t, q)$  データと、三角形の各頂点の  $(x, y)$  データとが、DDA データ S11 として、トライアングル DDA 回路 11 から



テクスチャエンジン回路 12 に出力される。

#### 【0041】

次に、テクスチャエンジン回路 12 およびメモリ I/F 回路 13 において、DDA データ S11 を用いて、「s/q」および「t/q」の算出処理、テクスチャ座標データ (u, v) の算出処理、テクスチャバッファ 20 からのデジタルデータとしての (R, G, B,  $\alpha$ ) データの読み出し処理、混合処理、および、ディスプレイバッファ 21 への書き込み処理が、図 3 に示す演算ブロック 200, 201, 202, , 203, 204, 205 でパイプライン方式で順に実行される。

#### 【0042】

次に、図 3 に示すテクスチャエンジン回路 12 およびメモリ I/F 回路 13 のパイプライン処理の動作について説明する。

ここでは、例えば、図 6 に示すような矩形 31 内の 8 画素について同時処理する場合を考える。この場合には、val データ S220<sub>1</sub>, S220<sub>2</sub>, S220<sub>3</sub>, S220<sub>5</sub>, S220<sub>6</sub> が「0」を示し、val データ S220<sub>4</sub>, S220<sub>7</sub>, S220<sub>8</sub> が「1」を示している。

#### 【0043】

val データ S220<sub>1</sub> ~ S220<sub>8</sub> および被演算データ S221<sub>1</sub> ~ S221<sub>8</sub> が、それぞれ対応する演算サブブロック 200<sub>1</sub> ~ 200<sub>8</sub> のクロックイネーブラ 210<sub>1</sub> ~ 210<sub>8</sub> に入力される。

そして、クロックイネーブラ 210<sub>1</sub> ~ 210<sub>8</sub> において、それぞれ val データ S220<sub>1</sub> ~ S220<sub>8</sub> のレベルが検出される。具体的には、クロックイネーブラ 210<sub>4</sub>, 210<sub>7</sub>, 210<sub>8</sub> において「1」が検出され、クロックイネーブラ 210<sub>1</sub>, 210<sub>2</sub>, 210<sub>3</sub>, 210<sub>5</sub>, 210<sub>6</sub> において「0」が検出される。

その結果、演算サブブロック 200<sub>4</sub>, 200<sub>7</sub>, 200<sub>8</sub> においてのみ、被演算データ S221<sub>4</sub>, S221<sub>7</sub>, S221<sub>8</sub> を用いて、「s/q」および「t/q」が算出され、当該除算結果 S200<sub>4</sub>, S200<sub>7</sub>, S200<sub>8</sub> が演算ブロック 201 の演算ブロック 201<sub>4</sub>, 201<sub>7</sub>, 201<sub>8</sub> に出力される。

一方、演算サブブロック $200_1$ 、 $200_2$ 、 $200_3$ 、 $200_5$ 、 $200_6$ では、除算は行なわれない。

また、除算結果 $S200_4$ 、 $S200_7$ 、 $S200_8$ の出力と同期して、 $val$ データ $S220_1 \sim S220_8$ が、演算ブロック201の演算サブブロック $201_1 \sim 201_8$ に出力される。

【0044】

次に、演算サブブロック $201_1 \sim 201_8$ のクロックイネーブラ $210_1 \sim 210_8$ において、それぞれ $val$ データ $S220_1 \sim S220_8$ のレベルが検出される。

そして、この検出結果に基づいて、演算サブブロック $201_4$ 、 $201_7$ 、 $201_8$ においてのみ、除算結果 $S200_4$ 、 $S200_7$ 、 $S200_8$ が示す「 $s/q$ 」および「 $t/q$ 」に、それぞれテクスチャサイズ $USIZE$ および $VSIZE$ を乗じて、テクスチャ座標データ $S202_4$ 、 $S202_7$ 、 $S202_8$ が生成され、それぞれ演算ブロック202の演算サブブロック $202_4$ 、 $202_7$ 、 $202_8$ に出力される。

一方、演算サブブロック $201_1$ 、 $201_2$ 、 $201_3$ 、 $201_5$ 、 $201_6$ では、演算は行なわれない。

また、テクスチャ座標データ $S202_4$ 、 $S202_7$ 、 $S202_8$ の出力と同期して、 $val$ データ $S220_1 \sim S220_8$ が、演算ブロック202の演算サブブロック $202_1 \sim 202_8$ に出力される。

【0045】

次に、演算サブブロック $202_1 \sim 202_8$ のクロックイネーブラ $212_1 \sim 212_8$ において、それぞれ $val$ データ $S220_1 \sim S220_8$ のレベルが検出される。

そして、この検出結果に基づいて、演算サブブロック $202_4$ 、 $202_7$ 、 $202_8$ においてのみ、SRAM17あるいはテクスチャバッファ20に記憶されているテクスチャデータの読み出し処理が行なわれ、 $(s, t)$ データに対応したテクスチャアドレスに記憶された $(R, G, B, \alpha)$ データが読み出される。

そして、この読み出した $(R, G, B, \alpha)$ データ $S202_4$ 、 $S202_7$ 、

S202<sub>8</sub> が、演算ブロック204の演算サブブロック203<sub>4</sub> , 203<sub>7</sub> , 203<sub>8</sub> に出力される。

一方、演算サブブロック202<sub>1</sub> , 202<sub>2</sub> , 202<sub>3</sub> , 202<sub>5</sub> , 202<sub>6</sub> では、読み出し処理は行なわれない。

また、(R, G, B,  $\alpha$ ) データS202<sub>4</sub> , S202<sub>7</sub> , S202<sub>8</sub> の出力と同期して、valデータS220<sub>1</sub> ~S220<sub>8</sub> が、演算ブロック203の演算サブブロック203<sub>1</sub> ~203<sub>8</sub> に出力される。

#### 【0046】

次に、演算サブブロック203<sub>1</sub> ~203<sub>8</sub> のクロックイネーブラ212<sub>1</sub> ~212<sub>8</sub> において、それぞれvalデータS220<sub>1</sub> ~S220<sub>8</sub> のレベルが検出される。

そして、この検出結果に基づいて、演算サブブロック203<sub>4</sub> , 203<sub>7</sub> , 203<sub>8</sub> においてのみ、それぞれ演算ブロック202から入力したテクスチャデータである(R, G, B,  $\alpha$ ) データS202<sub>4</sub> , 202<sub>7</sub> , 202<sub>8</sub> と、トライアングルDDA回路11からのDDAデータS11に含まれる(R, G, B) データとを、(R, G, B,  $\alpha$ ) データS202<sub>4</sub> , 202<sub>7</sub> , 202<sub>8</sub> に含まれる $\alpha$ データ(テクスチャ $\alpha$ ) が示す割合で混合し、(R, G, B) 混合データを生成する。

そして、演算サブブロック203<sub>4</sub> , 203<sub>7</sub> , 203<sub>8</sub> は、生成された(R, G, B) 混合データと、対応するDDAデータS11に含まれる $\alpha$ データとを含む(R, G, B,  $\alpha$ ) データS203<sub>4</sub> , 203<sub>7</sub> , 203<sub>8</sub> を、演算ブロック204に出力する。

一方、演算サブブロック203<sub>1</sub> , 203<sub>2</sub> , 203<sub>3</sub> , 203<sub>5</sub> , 203<sub>6</sub> では、混合処理は行なわれない。

#### 【0047】

次に、演算サブブロック204<sub>1</sub> ~204<sub>8</sub> のクロックイネーブラ214<sub>1</sub> ~214<sub>8</sub> において、それぞれvalデータS220<sub>1</sub> ~S220<sub>8</sub> のレベルが検出される。

そして、この検出結果に基づいて、演算サブブロック204<sub>4</sub> , 204<sub>7</sub> , 2

04<sub>8</sub> においてのみ、(R, G, B, α) データ S203<sub>4</sub>, S203<sub>7</sub>, S203<sub>8</sub> について、z バッファ 22 に記憶された z データの内容を用いて、z 比較が行なわれ、(R, G, B, α) データ S203<sub>4</sub>, S203<sub>7</sub>, S203<sub>8</sub> によって描画する画像が、前回、ディスプレイバッファ 21 に描画した値よりも手前に位置する場合には、z バッファ 22 が更新されると共に、(R, G, B, α) データ S203<sub>4</sub>, S203<sub>7</sub>, S203<sub>8</sub> が、それぞれ (R, G, B, α) データ S204<sub>4</sub>, S204<sub>7</sub>, S204<sub>8</sub> として、それぞれ演算サブブロック 205 の演算サブブロック 205<sub>4</sub>, 205<sub>7</sub>, 205<sub>8</sub> に出力される。

#### 【0048】

次に、演算サブブロック 205<sub>1</sub> ~ 205<sub>8</sub> のクロックイネーブラ 215<sub>1</sub> ~ 215<sub>8</sub> において、それぞれ val データ S220<sub>1</sub> ~ S220<sub>8</sub> のレベルが検出される。

そして、この検出結果に基づいて、(R, G, B, α) データ S204<sub>4</sub>, S204<sub>7</sub>, S204<sub>8</sub> の (R, G, B) データと、既にディスプレイバッファ 21 に記憶されている (R, G, B) データとが、α データが示す混合値で混合され、(R, G, B) データ S205<sub>4</sub>, S205<sub>7</sub>, S205<sub>8</sub> が最終的に算出される。

そして、この混合処理された、(R, G, B) データ S205<sub>4</sub>, S205<sub>7</sub>, S205<sub>8</sub> が、ディスプレイバッファ 21 に書き込まれる。

一方、演算サブブロック 204<sub>1</sub>, 204<sub>2</sub>, 204<sub>3</sub>, 204<sub>5</sub>, 204<sub>6</sub> では、混合処理は行なわれない。

#### 【0049】

すなわち、テクスチャエンジン回路 12 およびメモリ I/F 回路 13 では、図 6 に示す矩形 31 内の画素について同時に処理を行なう場合に、三角形 30 の外に位置する画素についての処理は行なわない。すなわち、図 4 に示す矩形 31 内の画素についての演算を行なっている間は、演算サブブロック 200<sub>1</sub>, 200<sub>2</sub>, 200<sub>3</sub>, 200<sub>5</sub>, 200<sub>6</sub>, 201<sub>1</sub>, 201<sub>2</sub>, 201<sub>3</sub>, 201<sub>5</sub>, 201<sub>6</sub>, 202<sub>1</sub>, 202<sub>2</sub>, 202<sub>3</sub>, 202<sub>5</sub>, 202<sub>6</sub>, 204<sub>1</sub>, 204<sub>2</sub>, 204<sub>3</sub>, 204<sub>5</sub>, 204<sub>6</sub>, 205<sub>1</sub>, 205<sub>2</sub>, 205<sub>3</sub>, 2

05<sub>5</sub> , 205<sub>6</sub> は停止した状態になり、これらの演算サブブロックは電力を消費しない。

#### 【0050】

以上説明したように、3次元コンピュータグラフィックシステム1によれば、テクスチャエンジン回路12におけるパイプライン処理において、同時処理する8画素のうち、処理対象となる三角形の外部に位置する画素についての演算は行なわないようにすることができる。

そのため、テクスチャエンジン回路12における消費電力を大幅に低減できる。その結果、3次元コンピュータグラフィックシステム1の電源として、簡単かつ安価なものをを用いることができる。

なお、テクスチャエンジン回路12は、図3および図4に示すように、各演算サブブロックに、クロックイネーブラおよび1ビットのフラグ用フリップフロップを組み込むことで、上述した機能を実現するが、クロックイネーブラおよび1ビットのフラグ用フリップフロップの回路規模は小さいため、テクスチャエンジン回路12の回路規模が大幅に増大することはない。

#### 【0051】

### 第2実施形態

図5は、本実施形態の3次元コンピュータグラフィックシステム451のシステム構成図である。

本実施形態の3次元コンピュータグラフィックシステム451は、 $\alpha$ ブレンド処理を行うか否かを各画素毎に予め判断し、 $\alpha$ ブレンド処理を行わないと判断した場合に、 $\alpha$ ブレンド処理を行う演算サブブロックのうち対応する演算サブブロックの処理を停止させる点を除いて、前述した第1実施形態の3次元コンピュータグラフィックシステム1と同じである。

すなわち、本実施形態では、各演算サブブロックは、第1実施形態の場合と同様に、対応する画素が処理対象となる三角形の外部に位置する場合には処理を停止する。また、演算サブブロックのうち $\alpha$ ブレンド処理を行う演算サブブロックは、対応する画素が処理対象となる三角形の外部に位置するか、あるいは対応する画素の $\alpha$ データが「0」である場合に処理を停止する。

【0052】

図5に示すように、3次元コンピュータグラフィックシステム451は、メインメモリ2、I/Oインタフェース回路3、メインプロセッサ4およびレンダリング回路425がメインバス6を介して接続されている。

図5において、図1と同じ符号を付した構成要素は、第1実施形態で説明した同一符号を付した構成要素と同じである。

すなわち、メインメモリ2、I/Oインタフェース回路3、メインプロセッサ4およびメインバス6は、第1実施形態で説明したものと同一である。

【0053】

また、図5に示すように、レンダリング回路425は、DDAセットアップ回路10、トライアングルDDA回路411、テクスチャエンジン回路12、メモリI/F回路413、CRTコントローラ回路14、RAMDAC回路15、DRAM16およびSRAM17を有する。

ここで、DDAセットアップ回路10、テクスチャエンジン回路12、CRTコントローラ回路14、RAMDAC回路15、DRAM16およびSRAM17は、第1実施形態で説明したものと同一である。

【0054】

以下、トライアングルDDA回路411およびメモリI/F回路413について説明する。

トライアングルDDA回路411

トライアングルDDA回路411は、前述した第1実施形態のトライアングルDDA回路11と同様に、DDAセットアップ回路10から入力した変分データS10を用いて、三角形内部の各画素の線形補間された(z, R, G, B, α, s, t, q)データを算出する。

トライアングルDDA回路411は、各画素の(x, y)データと、当該(x, y)座標の画素についての(z, R, G, B, α, s, t, q, val)データとを、DDAデータ(補間データ)S11としてテクスチャエンジン回路12に出力する。

本実施形態では、トライアングルDDA回路411は、並行して処理を行う矩

形内に位置する 8 画素分の DDA データ S11 を単位としてテクスチャエンジン回路 12 に出力する。

なお、以下、並行して処理を行う 8 画素についての (z, R, G, B,  $\alpha$ , s, t, q, val) データのうち、val データを val データ S220<sub>1</sub> ~ S220<sub>8</sub> とし、(z, R, G, B,  $\alpha$ , s, t, q) データを被演算データ S221<sub>1</sub> ~ S221<sub>8</sub> とする。

すなわち、トライアングル DDA 回路 11 は、8 画素分の (x, y) データと、val データ S220<sub>1</sub> ~ S220<sub>8</sub> と、被演算データ S221<sub>1</sub> ~ S221<sub>8</sub> とからなる DDA データ S11 をテクスチャエンジン回路 12 に出力する。

#### 【0055】

また、トライアングル DDA 回路 411 は、並行して処理を行う 8 画素について、上述したように線形補間して生成した (z, R, G, B,  $\alpha$ , s, t, q) データのうち  $\alpha$  データが「0」であるか否か、すなわち  $\alpha$  ブレンド処理を行うか否かを判断する。

そして、トライアングル DDA 回路 411 は、 $\alpha$  データが「0」であると判断した場合に、「0」（ $\alpha$  ブレンド処理を行わないことを）を示す val データ 411a<sub>1</sub> ~ S411a<sub>8</sub> をメモリ I/F 回路 413 に出力し、 $\alpha$  データが「0」ではないと判断した場合に、「1」（ $\alpha$  ブレンド処理を行うことを）を示す val データ 411a<sub>1</sub> ~ S411a<sub>8</sub> をメモリ I/F 回路 413 に出力する。

#### 【0056】

##### メモリ I/F 回路 413

図 6 は、テクスチャエンジン回路 12 およびメモリ I/F 回路 413 の構成図である。

図 6 に示すように、メモリ I/F 回路 413 は、演算ブロック 204 および演算ブロック 405 を有する。

なお、図 6 において、図 3 と同じ符号を付した構成要素は、第 1 実施形態で説明した同一符号を構成要素と同じである。

すなわち、テクスチャエンジン回路 12 は、第 1 実施形態で説明したものと同一であり、メモリ I/F 回路 413 の演算ブロック 204 も第 1 実施形態で説明

したものと同じである。

【0057】

以下、メモリ I/F 回路 413 の演算ブロック 405 について説明する。

〔演算ブロック 405〕

演算ブロック 405 は、演算サブブロック  $405_1 \sim 405_8$  を有し、演算サブブロック  $204_1 \sim 204_8$  から入力した (R, G, B,  $\alpha$ ) データ  $S204_1 \sim S204_8$  と、既にディスプレイバッファ 21 に記憶されている (R, G, B) データとを、それぞれ (R, G, B,  $\alpha$ ) データ  $S204_1 \sim S204_8$  に含まれる  $\alpha$  データが示す混合値で混合し、混合後の (R, G, B) データ  $S405_1 \sim S405_8$  をディスプレイバッファ 21 に書き込む (打ち込む)。

このとき、演算サブブロック  $405_1 \sim 405_8$  は、それぞれクロックイネーブラ  $415_1 \sim 415_8$  により、それぞれ演算ブロック 204 からの val データ  $S220_1 \sim S220_8$  および図 5 に示すトライアングル DDA 回路 411 からの val データ  $S411a_1 \sim S411a_8$  のレベルを検出し、双方のレベルが「1」の場合にのみ  $\alpha$  ブレンド処理を行う。

ここで、双方のレベルが「1」の場合とは、当該画素が処理対象となる三角形の内部に位置し、しかも、当該画素の  $\alpha$  データが「0」でない ( $\alpha$  ブレンド処理を行うことを示す) 場合である。

すなわち、演算サブブロック  $405_1 \sim 405_8$  は、それぞれ val データ  $S220_1 \sim S220_8$  および val データ  $S411a_1 \sim S411a_8$  のうちいずれか一方が「0」の場合には、 $\alpha$  ブレンド処理を行わない。

【0058】

なお、演算サブブロック  $405_1 \sim 405_8$  は、val データ  $S220_1 \sim S220_8$  のレベルが「1」であり、val データ  $S411a_1 \sim S411a_8$  のレベルが「0」の場合には、演算サブブロック  $204_1 \sim 204_8$  から入力した (R, G, B,  $\alpha$ ) データ  $S204_1 \sim S204_8$  をディスプレイバッファ 21 に書き込む。

【0059】

以下、3次元コンピュータグラフィックシステム 451 の動作について説明す



る。

3次元コンピュータグラフィックシステム451の全体動作は、基本的に前述した第1実施形態で説明した3次元コンピュータグラフィックシステム1の全体動作と同じである。

また、図6に示すテクスチャエンジン回路12およびメモリI/F回路413のパイプライン処理の動作は、演算ブロック200～204の処理については、前述した第1実施形態で説明した動作と同じである。

#### 【0060】

以下、演算ブロック405の動作について説明する。

それぞれ図6に示す演算サブブロック $204_1 \sim 204_8$ から演算サブブロック $415_1 \sim 415_8$ に、(R, G, B,  $\alpha$ )データ $S204_1 \sim S204_8$ およびvalデータ $S220_1 \sim S220_8$ が出力される。

また、図5に示すトライアングルDDA回路411において、線形補間して生成した(z, R, G, B,  $\alpha$ , s, t, q)データのうち $\alpha$ データが「0」であるか否かが判断され、当該判断の結果を示すvalデータ $411a_1 \sim S411a_8$ が図6に示す演算サブブロック $415_1 \sim 415_8$ にそれぞれ出力される。

そして、演算サブブロック $415_1 \sim 415_8$ において、それぞれクロックイネーブラ $415_1 \sim 415_8$ により、valデータ $S220_1 \sim S220_8$ およびvalデータ $S411a_1 \sim S411a_8$ のレベルが検出され、双方のレベルが「1」の場合にのみ $\alpha$ ブレンド処理が行われる。

$\alpha$ ブレンド処理では、(R, G, B,  $\alpha$ )データ $S204_1 \sim S204_8$ と、既にディスプレイバッファ21に記憶されている(R, G, B)データとが、それぞれ(R, G, B,  $\alpha$ )データ $S204_1 \sim S204_8$ に含まれる $\alpha$ データが示す混合値で混合されて(R, G, B)データ $S405_1 \sim S405_8$ が生成される。そして、(R, G, B)データ $S405_1 \sim S405_8$ が、ディスプレイバッファ21に書き込まれる。

#### 【0061】

すなわち、本実施形態では、演算サブブロック $415_1 \sim 415_8$ のそれぞれにおいて、valデータ $S220_1 \sim S220_8$ およびvalデータ $S411a$

$s_1 \sim s_{411ag}$  のうち何れか一方が「0」の場合には、 $\alpha$  ブレンド処理は行われない。

#### 【0062】

以上説明したように、3次元コンピュータグラフィックシステム451によれば、トライアングルDDA回路411において、各画素について $\alpha$  データが「0」であるか否かを判断する。

そして、メモリI/F回路413において、同時処理する8画素のうち処理対象となる三角形の内部に位置する画素であっても、トライアングルDDA回路411による上記判断の結果に基づいて、 $\alpha$  データが「0」の画素についての $\alpha$  ブレンド処理を行わないようにすることができる。

そのため、3次元コンピュータグラフィックシステム451によれば、前述した第1実施形態の3次元コンピュータグラフィックシステム1に比べてさらに、消費電力を低減できる。

#### 【0063】

##### 第3実施形態

図7は、本実施形態の3次元コンピュータグラフィックシステム551のシステム構成図である。

本実施形態の3次元コンピュータグラフィックシステム551では、例えば、処理対象となっている画素の $z$  データと $z$  バッファに記憶されている対応する $z$  データとの比較を行い、今回描画しようとする画像が前回描画した画像より奥側（視点側と反対の方向）にある場合には、当該画素についてのテクスチャ座標データ（ $u$ 、 $v$ ）の生成処理、テクスチャデータの読み出し処理、テクスチャ $\alpha$  ブレンド処理および $\alpha$  ブレンド処理を停止する。

#### 【0064】

図7に示すように、3次元コンピュータグラフィックシステム551は、メインメモリ2、I/Oインタフェース回路3、メインプロセッサ4およびレンダリング回路525がメインバス6を介して接続されている。

図7において、図1と同じ符号を付した構成要素は、第1実施形態で説明した同一符号を付した構成要素と同じである。

すなわち、メインメモリ 2、I/O インタフェース回路 3、メインプロセッサ 4 およびメインバス 6 は、第 1 実施形態で説明したものと同一である。

#### 【0065】

また、図 7 に示すように、レンダリング回路 525 は、DDA セットアップ回路 10、トライアングル DDA 回路 11、テクスチャエンジン回路 512、メモリ I/F 回路 513、CRT コントローラ回路 14、RAMDAC 回路 15、DRAM 16 および SRAM 17 を有する。

ここで、DDA セットアップ回路 10、トライアングル DDA 回路 11、CRT コントローラ回路 14、RAMDAC 回路 15、DRAM 16 および SRAM 17 は、第 1 実施形態で説明したものと同一である。

#### 【0066】

以下、テクスチャエンジン回路 512 およびメモリ I/F 回路 513 について説明する。

図 8 は、テクスチャエンジン回路 512 およびメモリ I/F 回路 513 の構成図である。

図 8 に示すように、テクスチャエンジン回路 512 は、演算ブロック 500、501、502、503、504 を有する。

また、メモリ I/F 回路 513 は、演算ブロック 505 を有する。

本実施形態では、演算ブロック 500～505 は、それぞれ 8 画素についての処理を同時に行い、パイプライン処理が行われるように直列に接続されている。

ここで、演算ブロック 500 では  $z$  比較処理が行われ、演算ブロック 501 では「 $s/q$ 」および「 $t/q$ 」の算出処理が行われ、演算ブロック 502 ではテクスチャ座標データ ( $u, v$ ) の算出処理が行われ、演算ブロック 503 ではテクスチャバッファ 20 からの ( $R, G, B, \alpha$ ) データの読み出し処理が行われ、演算ブロック 504 ではテクスチャ  $\alpha$  ブレンド処理が行われ、演算ブロック 505 では  $\alpha$  ブレンド処理が行われる。

#### 【0067】

##### 〔演算ブロック 500〕

演算ブロック 500 は、演算サブブロック 500<sub>1</sub>～500<sub>8</sub> を有し、図 7 に

示すトライアングルDDA回路11からDDAデータS11を入力する。

演算サブブロック $500_1 \sim 500_8$ は、それぞれクロックイネーブラ $214_1 \sim 214_8$ において、DDAデータS11に含まれるvalデータ $S220_1 \sim S220_8$ のレベル検出を行い、その結果、当該レベルが「1」の場合（当該画素が、処理対象となる三角形の内部に位置する場合）にはz比較処理を行い、当該レベルが「1」でない場合にはz比較処理を行わない。

#### 【0068】

演算サブブロック $500_1 \sim 500_8$ は、z比較処理において、DDAデータS11に含まれる被演算データ $S221_1 \sim S221_8$ のzデータと、zバッファ22に記憶された対応するzデータとを比較する。

そして、演算サブブロック $500_1 \sim 500_8$ は、被演算データ $S221_1 \sim S221_8$ によって描画する画像が、前回、ディスプレイバッファ21に描画した値よりも手前（視点側）に位置する場合には、それぞれ「1」を示すvalデータ $S500a_1 \sim S500a_8$ を演算ブロック501の演算サブブロック $501_1 \sim 501_8$ に出力し、それぞれ被演算データ $S221_1 \sim S221_8$ のzデータで、zバッファ22に記憶されている対応するzデータを書き換える。このとき、演算サブブロック $500_1 \sim 500_8$ は、さらに被演算データ $S221_1 \sim S221_8$ を演算サブブロック $501_1 \sim 501_8$ に出力する。

一方、演算サブブロック $500_1 \sim 500_8$ は、被演算データ $S221_1 \sim S221_8$ によって描画する画像が、前回、ディスプレイバッファ21に描画した値よりも手前（視点側）に位置しない場合には、それぞれ「0」を示すvalデータ $S500a_1 \sim S500a_8$ を演算ブロック501の演算サブブロック $501_1 \sim 501_8$ に出力し、zバッファ22に記憶されている対応するzデータを書き換えない。

#### 【0069】

##### 〔演算ブロック501〕

演算ブロック501は、DDAデータS11が示す（s，t，q）データを用いて、sデータをqデータで除算する演算と、tデータをqデータで除算する演算とを行う。

演算ブロック501は、図8に示すように、8個の演算サブブロック501<sub>1</sub>～501<sub>8</sub>を内蔵する。

ここで、演算サブブロック501<sub>1</sub>は、被演算データS221<sub>1</sub>およびvalデータS220<sub>1</sub>、S500a<sub>1</sub>を入力し、クロックイネーブラ511<sub>1</sub>～511<sub>8</sub>により、valデータS220<sub>1</sub>およびS500a<sub>1</sub>の双方が「1」、すなわち有効であるか否かを判断し、双方が「1」と判断した場合に、「s/q」および「t/q」を算出し、これを除算結果S501<sub>1</sub>として演算ブロック502の演算サブブロック502<sub>1</sub>に出力する。

#### 【0070】

また、演算サブブロック501<sub>1</sub>は、valデータS220<sub>1</sub>およびS500a<sub>1</sub>のいずれか一方が「0」、すなわち無効であることを示すと判断した場合には演算は行わず、除算結果S501<sub>1</sub>を出力しないか、あるいは、所定の仮値を示す除算結果S501<sub>1</sub>を演算ブロック502の演算サブブロック502<sub>1</sub>に出力する。

なお、演算サブブロック501<sub>2</sub>～501<sub>8</sub>も、それぞれ対応する画素について、演算サブブロック501<sub>1</sub>と同じ演算を行い、それぞれ除算結果S501<sub>2</sub>～S501<sub>8</sub>を後段の演算ブロック502の演算サブブロック502<sub>2</sub>～502<sub>8</sub>にそれぞれ出力する。

#### 【0071】

##### 〔演算ブロック502〕

演算ブロック502は、演算サブブロック502<sub>1</sub>～502<sub>8</sub>を有し、演算ブロック501から入力した除算結果S501<sub>1</sub>～S501<sub>8</sub>が示す「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEを乗じて、テクスチャ座標データ(u, v)を生成する。

演算サブブロック502<sub>1</sub>は、クロックイネーブラ512<sub>1</sub>においてvalデータS220<sub>1</sub>およびS500a<sub>1</sub>のレベル検出を行い、双方のレベルが「1」の場合にのみ演算を行い、それぞれ演算結果であるテクスチャ座標データS502<sub>1</sub>を、演算ブロック503の演算サブブロック503<sub>1</sub>に出力する。

演算サブブロック502<sub>2</sub>～502<sub>8</sub>も、演算サブブロック502<sub>1</sub>と同様に

、対応するデータの処理を行う。

【0072】

〔演算ブロック503〕

演算ブロック503は、演算サブブロック $503_1 \sim 503_8$ を有し、メモリI/F回路13を介して、SRAM17あるいはDRAM16に、演算ブロック502で生成したテクスチャ座標データ(u, v)を含む読み出し要求を出力し、メモリI/F回路13を介して、SRAM17あるいはテクスチャバッファ20に記憶されているテクスチャデータを読み出すことで、(u, v)データに対応したテクスチャアドレスに記憶された(R, G, B,  $\alpha$ )データS17を得る。

演算サブブロック $503_1$ は、クロックイネーブラ $513_1$ においてvalデータ $S220_1$ および $S500a_1$ のレベル検出を行い、双方のレベルが「1」の場合にのみ読み出し処理を行い、それぞれ読み出した(R, G, B,  $\alpha$ )データS17を、(R, G, B,  $\alpha$ )データ $S503_1$ として、演算ブロック203の演算サブブロック $504_1$ に出力する。

演算サブブロック $503_2 \sim 503_8$ も、演算サブブロック $503_1$ と同様に、対応するデータの処理を行う。

【0073】

〔演算ブロック504〕

演算ブロック504は、演算サブブロック $504_1 \sim 504_8$ を有し、演算ブロック503から入力したテクスチャデータである(R, G, B,  $\alpha$ )データ $S503_1 \sim S503_8$ と、トライアングルDDA回路11からの対応するDDAデータS11に含まれる(R, G, B)データとを、(R, G, B,  $\alpha$ )データ $S503_1 \sim S503_8$ に含まれる $\alpha$ データ(テクスチャ $\alpha$ )が示す割合で混合し、(R, G, B)混合データを生成する。

そして、演算ブロック504は、生成された(R, G, B)混合データと、対応するDDAデータS11に含まれる $\alpha$ データとを含む(R, G, B,  $\alpha$ )データ $S504_1 \sim S504_8$ を、演算ブロック505に出力する。

演算サブブロック $504_1 \sim 504_8$ は、それぞれクロックイネーブラ $514$

$1 \sim 514_8$  により  $val$  データ  $S220_1 \sim S220_8$  および  $S500a_1 \sim S500a_8$  のレベル検出を行い、双方のレベルが「1」の場合にのみ上記混合処理を行う。

#### 【0074】

##### 〔演算ブロック505〕

演算ブロック505は、演算サブブロック  $505_1 \sim 505_8$  を有し、入力した  $(R, G, B, \alpha)$  データ  $S504_1 \sim S504_8$  と、既にディスプレイバッファ21に記憶されている  $(R, G, B)$  データとを、それぞれ  $(R, G, B, \alpha)$  データ  $S504_1 \sim S504_8$  に含まれる  $\alpha$  データが示す混合値で混合し、混合後の  $(R, G, B)$  データ  $S505_1 \sim S505_8$  をディスプレイバッファ21に書き込む（打ち込む）。

演算サブブロック  $505_1 \sim 505_8$  は、それぞれクロックイネーブラ  $215_1 \sim 215_8$  において  $val$  データ  $S220_1 \sim S220_8$  および  $S500a_1 \sim S500a_8$  のレベルを検出し、双方のレベルが「1」の場合にのみ上記混合処理およびディスプレイバッファ21への書き込み処理を行う。

#### 【0075】

以下、図8に示すテクスチャエンジン回路512およびメモリI/F回路513のパイプライン処理の動作について説明する。

まず、演算サブブロック  $500_1 \sim 500_8$  のクロックイネーブラ  $214_1 \sim 214_8$  において、それぞれDDAデータ  $S11$  に含まれる  $val$  データ  $S220_1 \sim S220_8$  のレベル検出が行われ、当該レベルが「1」の場合（当該画素が、処理対象となる三角形の内部に位置する場合）には  $z$  比較処理が行われる。

そして、被演算データ  $S221_1 \sim S221_8$  によって描画する画像が、前回、ディスプレイバッファ21に描画した値よりも手前（視点側）に位置する場合には、それぞれ「1」を示す  $val$  データ  $S500a_1 \sim S500a_8$  が演算ブロック501の演算サブブロック  $501_1 \sim 501_8$  に出力され、それぞれ被演算データ  $S221_1 \sim S221_8$  の  $z$  データで、 $z$  バッファ22に記憶されている対応する  $z$  データが書き換えられる。このとき、さらに被演算データ  $S221_1 \sim S221_8$  が、演算サブブロック  $500_1 \sim 500_8$  から演算サブブロック

501<sub>1</sub> ~ 501<sub>8</sub> に出力される。

一方、valデータS220<sub>1</sub> ~ S220<sub>8</sub> のレベルが「1」でない場合にはz比較処理は行われず、それぞれ「0」を示すvalデータS500a<sub>1</sub> ~ S500a<sub>8</sub> が演算ブロック501の演算サブブロック501<sub>1</sub> ~ 501<sub>8</sub> に出力される。このとき、zバッファ22に記憶されている対応するzデータは書き換えられない。

【0076】

次に、演算サブブロック501<sub>1</sub> ~ 501<sub>8</sub> のクロックイネーブラ511<sub>1</sub> ~ 511<sub>8</sub> において、valデータS220<sub>1</sub> およびS500a<sub>1</sub> の双方が「1」、すなわち有効であるか否かが判断され、双方が「1」であると判断された場合に、「s/q」および「t/q」が算出され、これが除算結果S501<sub>1</sub> ~ S501<sub>8</sub> として演算ブロック502の演算サブブロック502<sub>1</sub> ~ 502<sub>8</sub> に出力される。

一方、valデータS220<sub>1</sub> ~ S220<sub>8</sub> およびS500a<sub>1</sub> ~ S500a<sub>8</sub> のいずれか一方が「0」、すなわち無効であることを示すと判断された場合には、それぞれ演算サブブロック501<sub>1</sub> ~ 501<sub>8</sub> では演算は行われず。

【0077】

次に、演算サブブロック502<sub>1</sub> ~ 502<sub>8</sub> のクロックイネーブラ512<sub>1</sub> ~ 512<sub>8</sub> においてvalデータS220<sub>1</sub> ~ S220<sub>8</sub> およびS500a<sub>1</sub> ~ S500a<sub>8</sub> のレベル検出が行われる。

そして、双方のレベルが「1」の場合にのみ、演算サブブロック502<sub>1</sub> ~ 502<sub>8</sub> において、それぞれ演算ブロック501から入力した除算結果S501<sub>1</sub> ~ S501<sub>8</sub> が示す「s/q」および「t/q」に、それぞれテクスチャサイズUSIZEおよびVSIZEが乗算され、テクスチャ座標データ(u, v)が生成される。テクスチャ座標データ(u, v)は、それぞれ演算サブブロック503<sub>1</sub> ~ 503<sub>8</sub> に出力される。

【0078】

次に、演算サブブロック503<sub>1</sub> ~ 503<sub>8</sub> のクロックイネーブラ513<sub>1</sub> ~ 513<sub>8</sub> において、valデータS220<sub>1</sub> ~ S220<sub>8</sub> およびS500a<sub>1</sub> ~



S500a<sub>g</sub> のレベル検出が行われ、双方のレベルが「1」の場合にのみ、テクスチャ座標データ (u, v) を含む読み出し要求がSRAM17に出力され、メモリI/F回路13を介してテクスチャデータが読み出され、(u, v) データに対応したテクスチャアドレスに記憶された (R, G, B, α) データS17が得られる。(R, G, B, α) データS17は、(R, G, B, α) データS503<sub>1</sub> ~ S503<sub>8</sub> として、演算サブブロック504<sub>1</sub> ~ 504<sub>8</sub> に出力される。

#### 【0079】

次に、演算サブブロック504<sub>1</sub> ~ 504<sub>8</sub> のクロックイネーブラ514<sub>1</sub> ~ 514<sub>8</sub> によりvalデータS220<sub>1</sub> ~ S220<sub>8</sub> およびS500a<sub>1</sub> ~ S500a<sub>g</sub> のレベル検出が行われ、双方のレベルが「1」の場合にのみ、(R, G, B, α) データS503<sub>1</sub> ~ S503<sub>8</sub> と、トライアングルDDA回路11からの対応するDDAデータS11に含まれる (R, G, B) データとが、(R, G, B, α) データS503<sub>1</sub> ~ S503<sub>8</sub> に含まれるαデータ (テクスチャα) が示す割合で混合され、(R, G, B) 混合データが生成される。

そして、当該生成された (R, G, B) 混合データと、対応するDDAデータS11に含まれるαデータとを含む (R, G, B, α) データS504<sub>1</sub> ~ S504<sub>8</sub> が、演算サブブロック504<sub>1</sub> ~ 504<sub>8</sub> から演算サブブロック505<sub>1</sub> ~ 505<sub>8</sub> に出力される。

#### 【0080】

次に、演算サブブロック505<sub>1</sub> ~ 505<sub>8</sub> のクロックイネーブラ215<sub>1</sub> ~ 215<sub>8</sub> において、valデータS220<sub>1</sub> ~ S220<sub>8</sub> およびS500a<sub>1</sub> ~ S500a<sub>g</sub> のレベルが検出され、双方のレベルが「1」の場合にのみ、(R, G, B, α) データS504<sub>1</sub> ~ S504<sub>8</sub> と、既にディスプレイバッファ21に記憶されている (R, G, B) データとが、それぞれ (R, G, B, α) データS504<sub>1</sub> ~ S504<sub>8</sub> に含まれるαデータが示す混合値で混合され、混合後の (R, G, B) データS505<sub>1</sub> ~ S505<sub>8</sub> がディスプレイバッファ21に書き込まれる。

## 【0081】

以上説明したように、3次元コンピュータグラフィックシステム551によれば、テクスチャエンジン回路512の初段の演算ブロック500において各画素に関する $z$ 比較処理を行い、後の処理によって生成される画像データがディスプレイバッファ21に書き込まれるものであるかを判断する。

そして、テクスチャエンジン回路512およびメモリI/F回路513において、同時処理する8画素のうち処理対象となる三角形の内部に位置する画素であっても、演算ブロック500による上記判断の結果に基づいて、ディスプレイバッファ21に書き込まない画像データに関する処理を行わないように（停止）する。

そのため、3次元コンピュータグラフィックシステム551によれば、前述した第1実施形態の3次元コンピュータグラフィックシステム1に比べてさらに、消費電力を低減できる。

## 【0082】

本発明は上述した実施形態には限定されない。

例えば、前述した第2実施形態では、図6に示すように、テクスチャエンジン回路12およびメモリI/F回路413の各演算ブロックで8画素のデータについて同時に処理する場合について例示したが、図9に示すように、各演算ブロックで1画素のデータの処理を行うようにしてもよい。

この場合には、処理対象となる画素の被演算データ $S221_1$ のみがテクスチャエンジン回路12に入力されるため、 $val$ データ $S220_1$ は不要となる。すなわち、演算サブブロック $200_1$ 、 $201_1$ 、 $202_1$ 、 $203_1$ 、 $204_1$ では常に演算が行われ、演算サブブロック $405_1$ では $val$ データ $S400a_1$ のレベルが「1」の場合にのみ $\alpha$ ブレンド処理が行われる。

## 【0083】

また、前述した第3実施形態では、図8に示すように、テクスチャエンジン回路512およびメモリI/F回路513の各演算ブロックで8画素のデータについて同時に処理する場合について例示したが、図10に示すように、各演算ブロックで1画素のデータの処理を行うようにしてもよい。

この場合には、処理対象となる画素の被演算データ  $S221_1$  のみがテクスチャエンジン回路 512 に入力されるため、 $val$  データ  $S220_1$  は不要となる。すなわち、演算サブブロック  $500_1$  では  $z$  比較処理が常に行われ、演算サブブロック  $501_1$ ,  $502_1$ ,  $503_1$ ,  $504_1$ ,  $505_1$  では、演算サブブロック  $500_1$  で生成された  $val$  データ  $S500a_1$  のレベルが「1」の場合にのみ処理が行われる。

## 【0084】

また、例えば、上述した実施形態では、図 3 に示すように、テクスチャエンジン回路 12 およびメモリ I/F 回路 13 におけるパイプライン処理を行なう演算サブブロックについて、 $val$  データ  $S220_1 \sim S220_8$  を利用する場合を例示したが、例えば、図 1 にレンダリング回路 5 内の DDA セットアップ回路 10、トライアングル DDA 回路 11、テクスチャエンジン回路 12 およびメモリ I/F 回路 13 における処理のうち、パイプライン処理を行わない所定の処理について、図 11 に示すように、 $val$  データ  $S320_1 \sim S320_8$  を用いて、演算処理の実行の有無を決定するようにしてもよい。

## 【0085】

また、上述した実施形態では、SRAM 17 を用いる構成を例示したが、SRAM 17 を設けない構成にしてもよい。

また、テクスチャバッファ 20 およびテクスチャ CLUT バッファ 23 を、DRAM 16 の外部に設けてもよい。

## 【0086】

また、上述した実施形態では、3次元画像を表示する場合を例示したが、本発明は複数画素についてのデータを同時に処理して2次元画像を表示する場合にも適用できる。

また、上述した実施形態では、図 2 に示すように、画像処理の対象となる ( $z$ ,  $R$ ,  $G$ ,  $B$ ,  $\alpha$ ,  $s$ ,  $t$ ,  $q$ ) データに、有効指示データとしての  $val$  データを付加した DDA データ  $S11$  を用いた場合を例示したが、( $z$ ,  $R$ ,  $G$ ,  $B$ ,  $\alpha$ ,  $s$ ,  $t$ ,  $q$ ) データと、 $val$  データとを別個独立のデータとして扱うようにしてもよい。

【 0 0 8 7 】

また、上述した実施形態では、ポリゴンレンダリングデータを生成するジオメトリ処理を、メインプロセッサ 4 で行なう場合を例示したが、レンダリング回路 5 で行なう構成にしてもよい。

【 0 0 8 8 】

さらに、上述した実施形態では、単位図形として三角形を例示したが、単位図形は特に限定されず、例えば、矩形であってもよい。

【 0 0 8 9 】

【発明の効果】

以上説明したように、本発明の画像処理装置およびその方法によれば、消費電力の大幅な低下を図ることができる。

そのため、本発明の画像処理装置によれば、小規模かつ簡単な構成の電源を用いることができ、小規模化が図れる。

【図面の簡単な説明】

【図 1】

図 1 は、本発明の第 1 実施形態の 3 次元コンピュータグラフィックシステムのシステム構成図である。

【図 2】

図 1 に示すトライアングル DDA 回路から出力される DDA データのフォーマットを説明するための図である。

【図 3】

図 3 は、図 1 に示すテクスチャエンジン回路およびメモリ I / F 回路の部分構成図である。

【図 4】

図 4 は、図 3 に示す演算サブブロックの内部構成図である。

【図 5】

図 5 は、本発明の第 2 実施形態の 3 次元コンピュータグラフィックシステムのシステム構成図である。

【図 6】

図 6 は、図 5 に示すテクスチャエンジン回路およびメモリ I / F 回路の部分構成図である。

【図 7】

図 7 は、本発明の第 3 実施形態の 3 次元コンピュータグラフィックシステムのシステム構成図である。

【図 8】

図 8 は、図 7 に示すテクスチャエンジン回路およびメモリ I / F 回路の部分構成図である。

【図 9】

図 9 は、図 5 に示す 3 次元コンピュータグラフィックシステムの変形例の構成図である。

【図 1 0】

図 1 0 は、図 7 に示す 3 次元コンピュータグラフィックシステムの変形例の構成図である。

【図 1 1】

図 1 1 は、図 1 に示す 3 次元コンピュータグラフィックシステムにおけるクロックイネーブラーを適用した、パイプライン処理を行っていない演算ブロックの構成図である。

【図 1 2】

図 1 2 は、従来技術の問題点を説明するための図である。

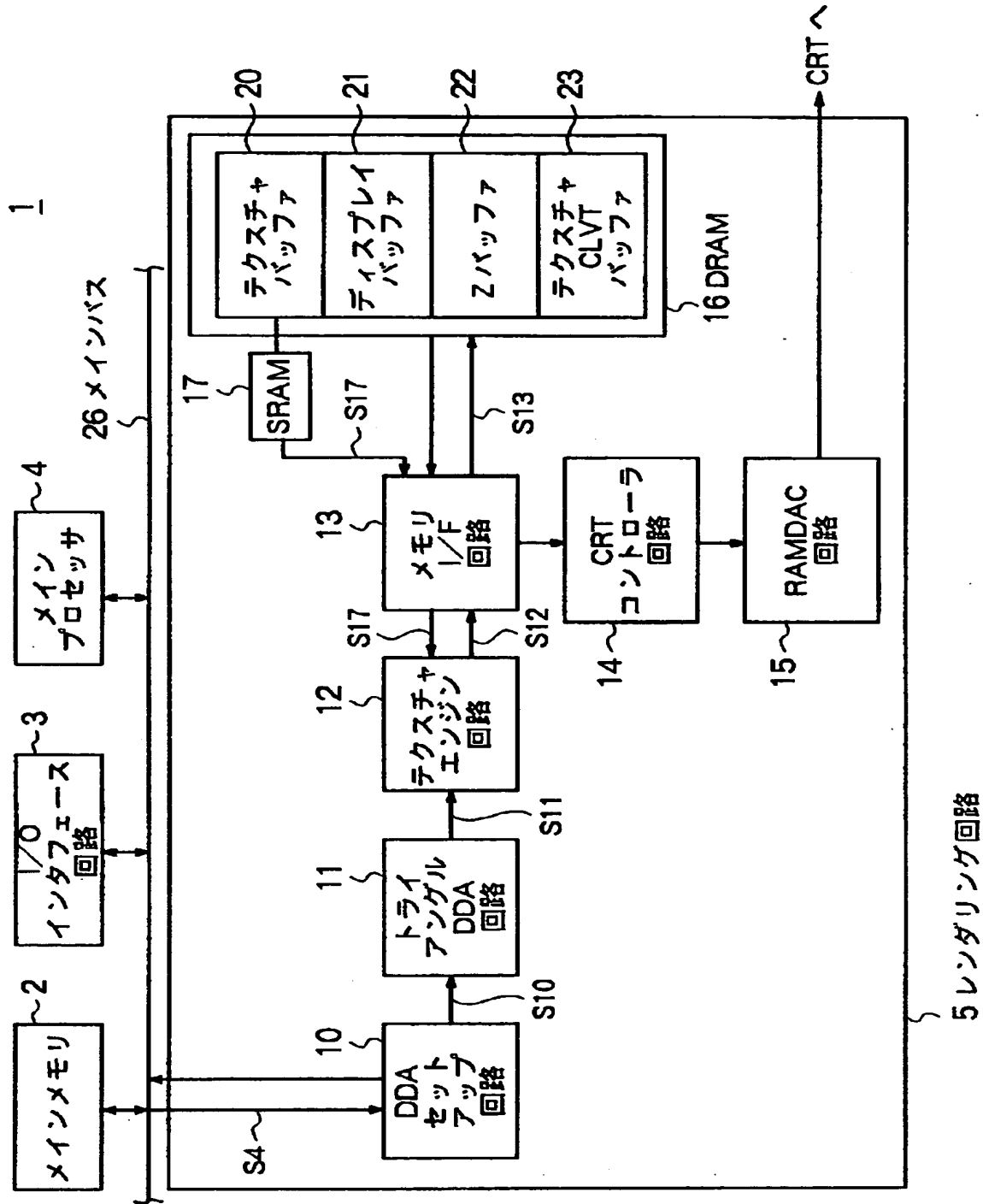
【符号の説明】

1 … 3 次元コンピュータグラフィックシステム、2 … メインメモリ、3 … I / O インタフェース回路、4 … メインプロセッサ、5 … レンダリング回路、1 0 … DDA セットアップ回路、1 1 … トライアングル DDA 回路、1 2 … テクスチャエンジン回路、1 3 … メモリ I / F 回路、1 4 … CRT コントローラ回路、1 5 … RAMDAC 回路、1 6 … DRAM、1 7 … SRAM、2 0 … テクスチャバッファ、2 1 … ディスプレイバッファ、2 2 … Z バッファ、2 3 … テクスチャ CLUT バッファ、2 0 0 ~ 2 0 5 … 演算ブロック、 $200_1 \sim 200_8$ 、 $201_1$

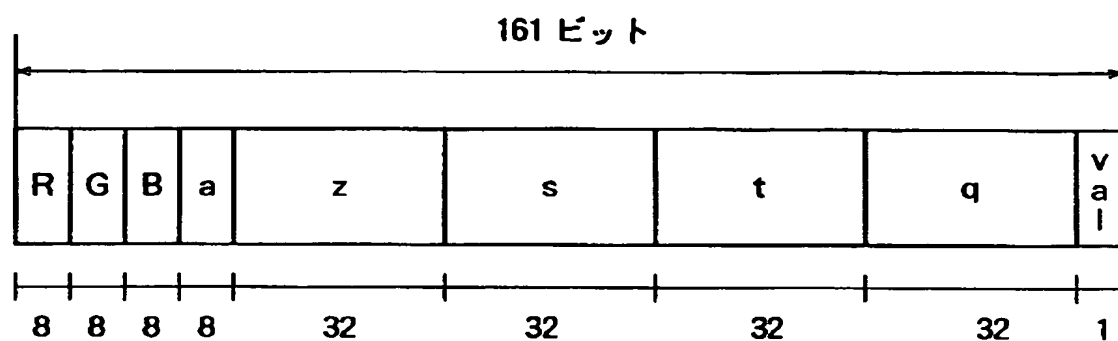
～201<sub>8</sub>，202<sub>1</sub>～202<sub>8</sub>，203<sub>1</sub>～203<sub>8</sub>，204<sub>1</sub>～204<sub>8</sub>，  
205<sub>1</sub>～205<sub>8</sub>…演算サブブロック、210<sub>1</sub>～210<sub>8</sub>，211<sub>1</sub>～21  
1<sub>8</sub>，212<sub>1</sub>～212<sub>8</sub>，213<sub>1</sub>～213<sub>8</sub>，214<sub>1</sub>～214<sub>8</sub>，215  
1～215<sub>8</sub>…クロックイネーブラ、222…データ用フリップフロップ、22  
3…プロセッサエレメント、224…フラグ用フリップフロップ

【書類名】 図面

【図 1】

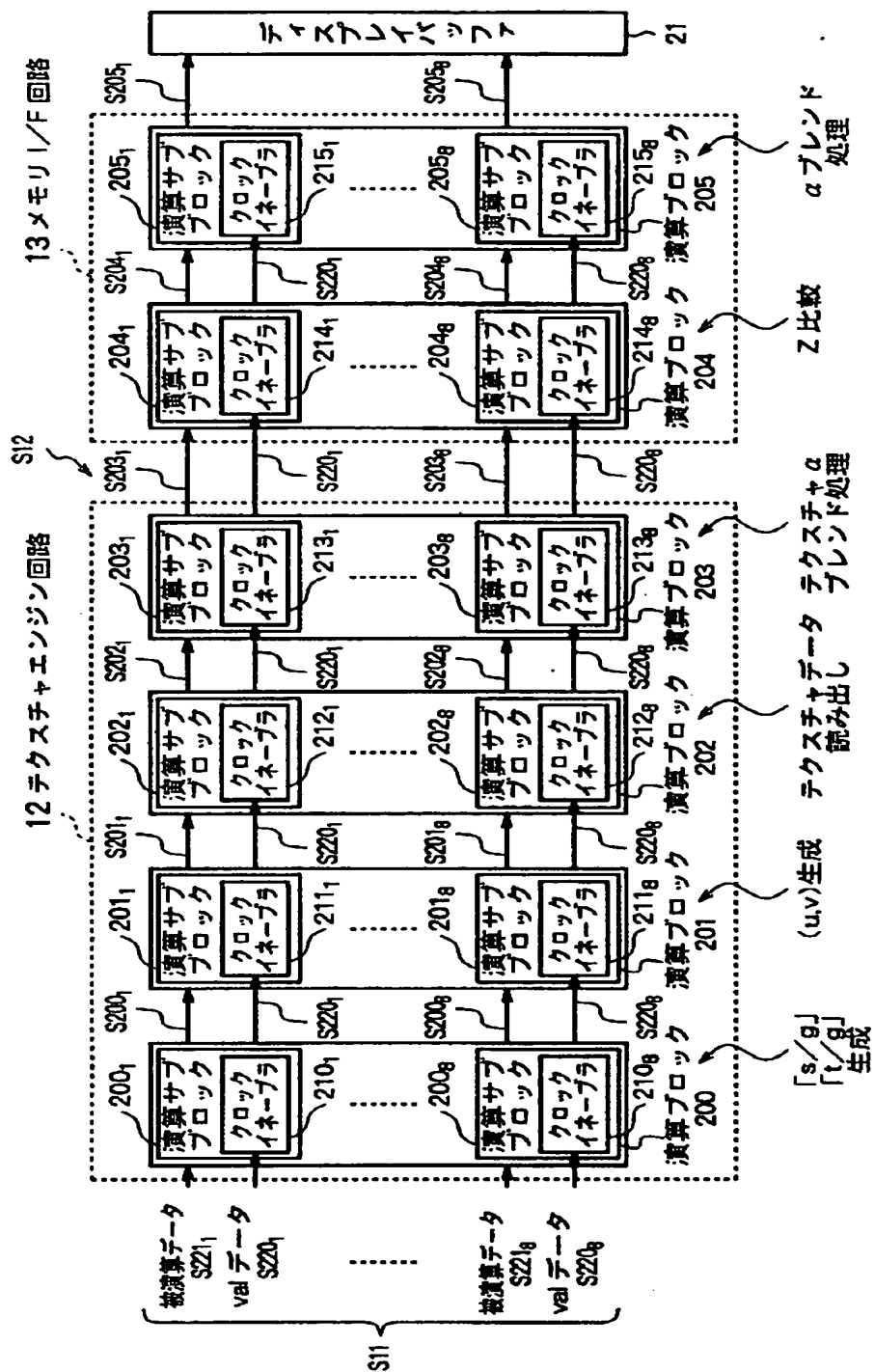


【図 2】

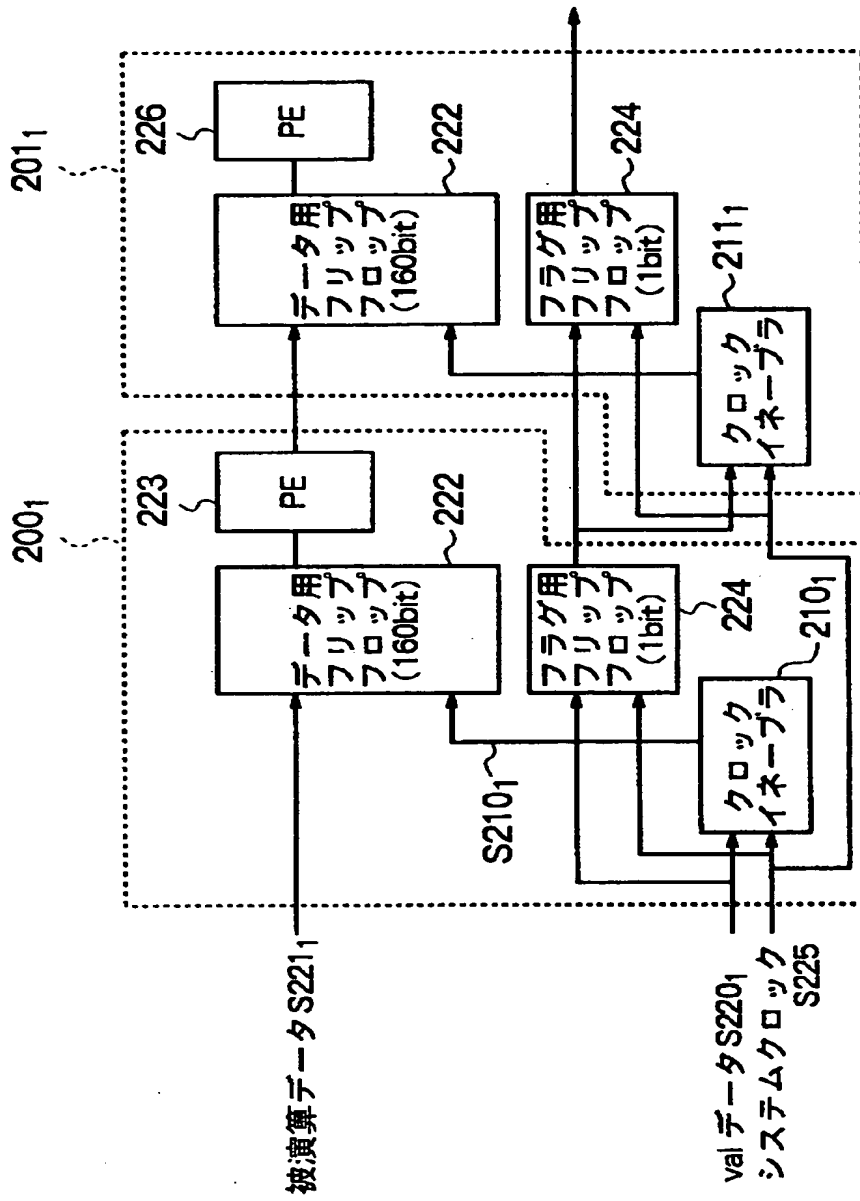




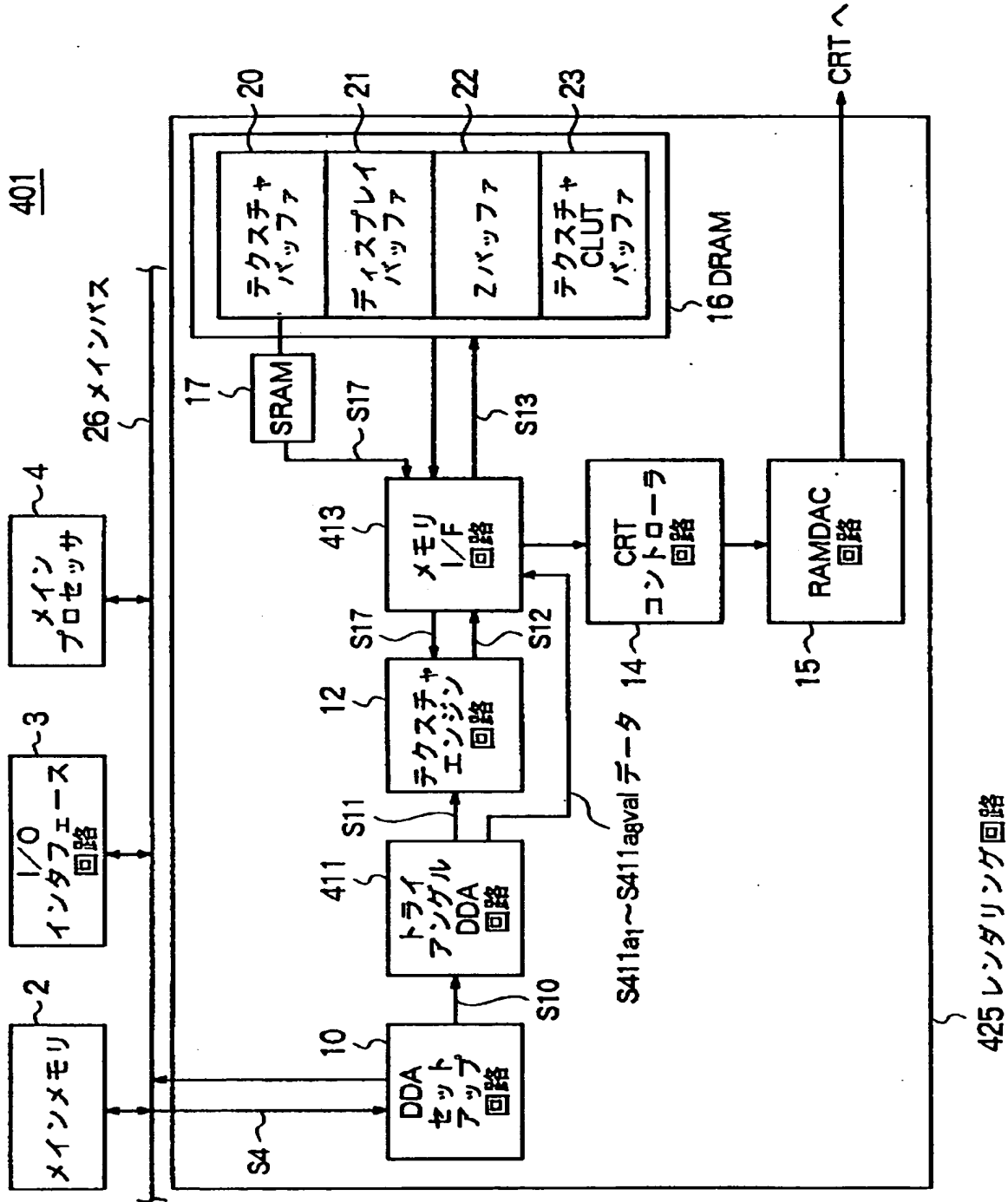
【図 3】



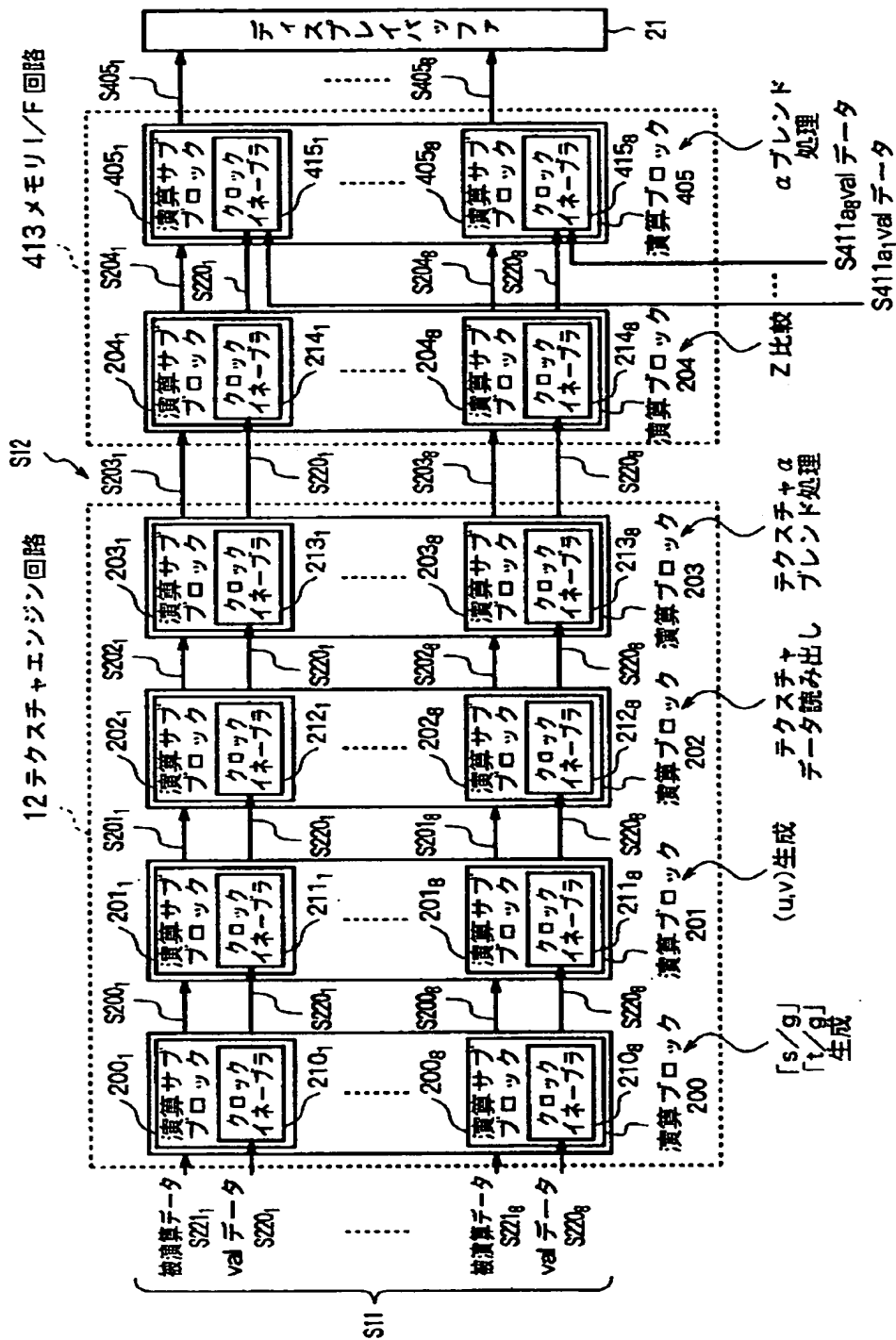
【図 4】



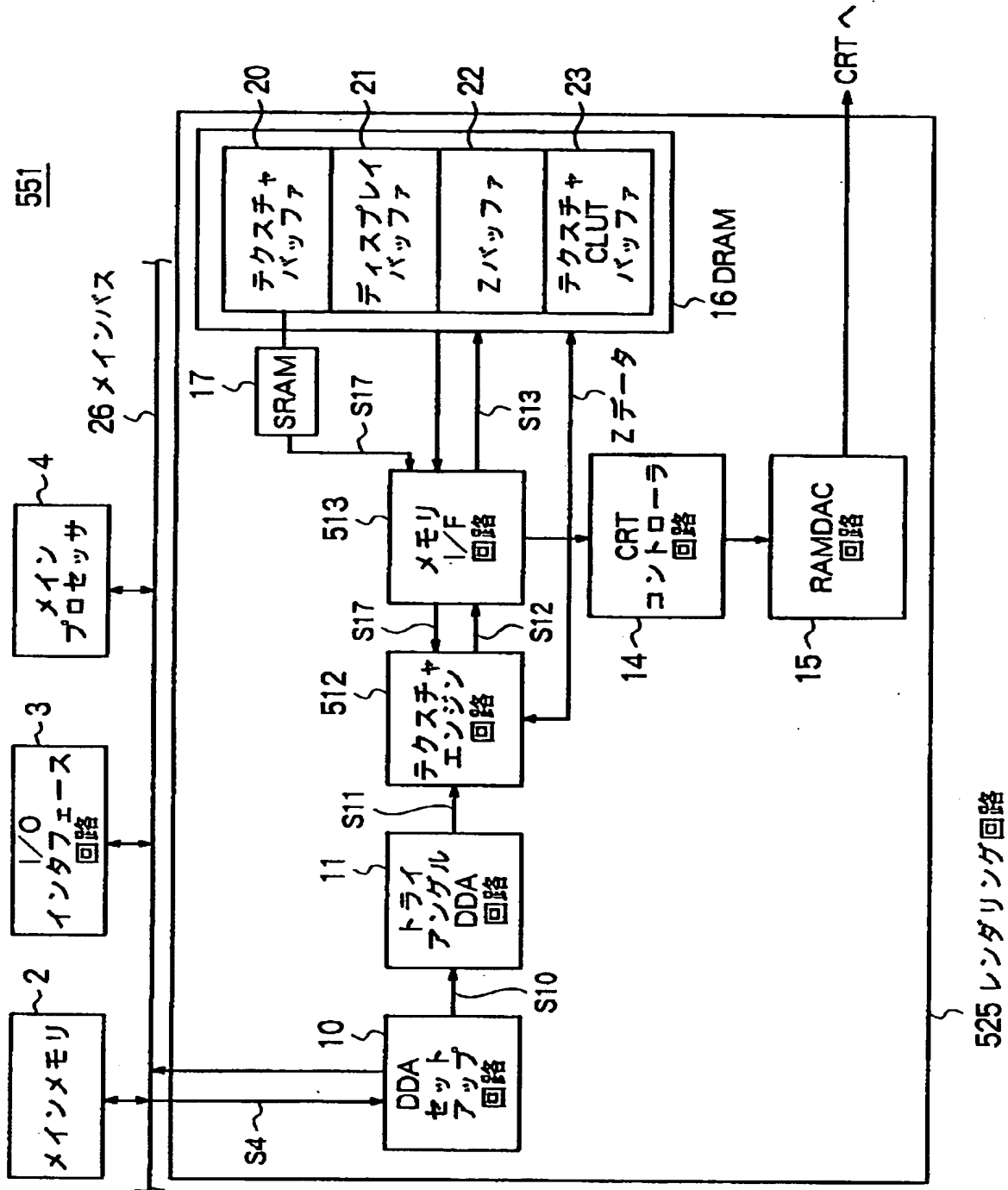
【図 5】



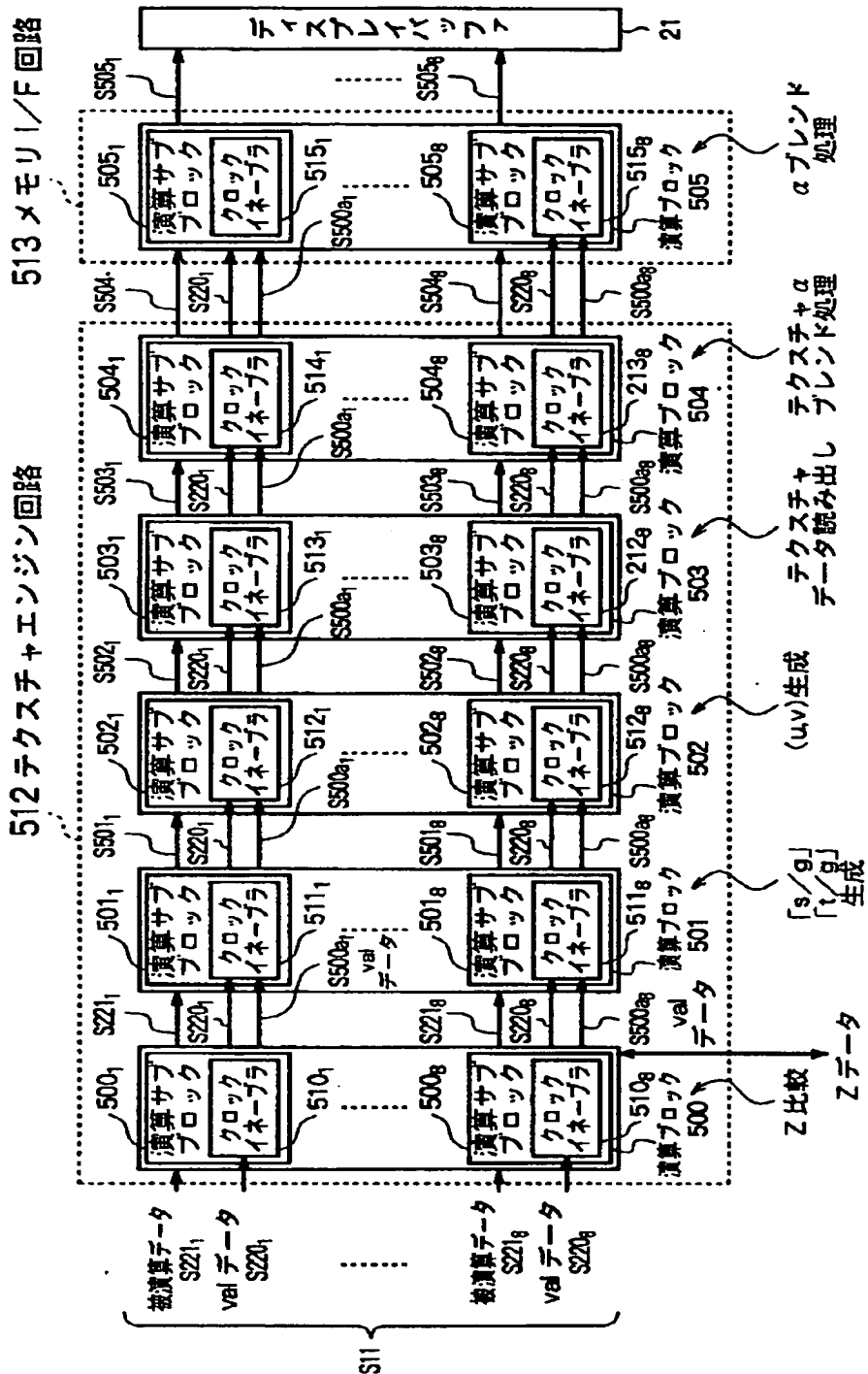
【図 6】



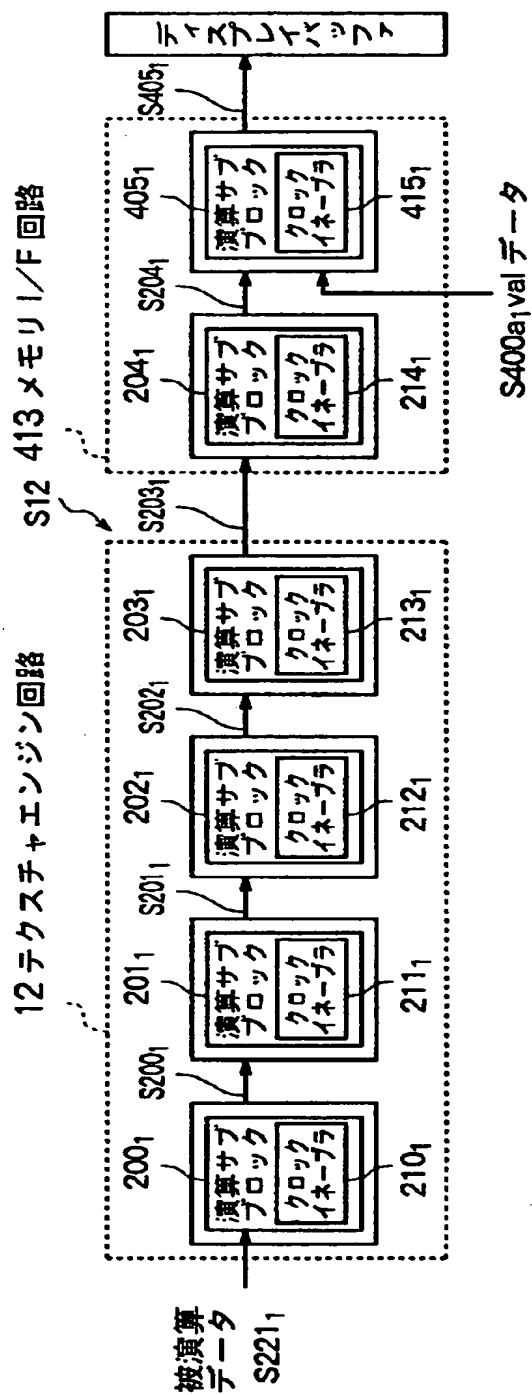
【図 7】



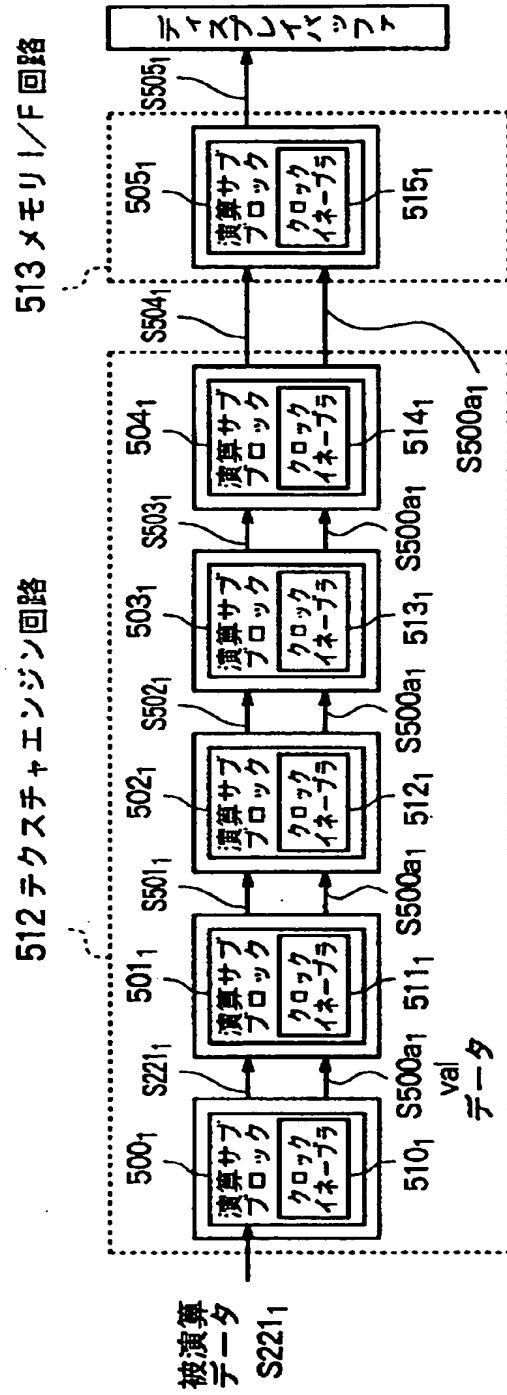
【图 8】



【図 9】

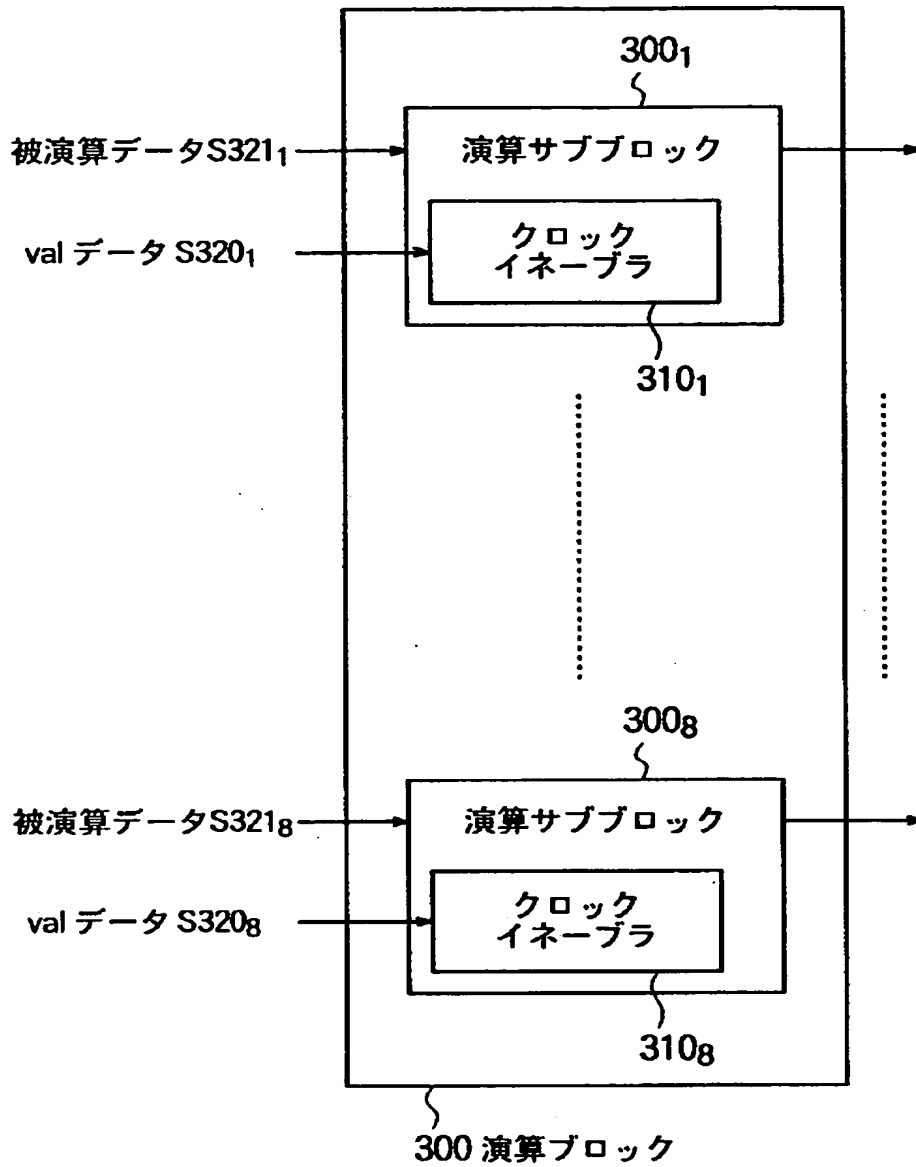


【図 10】

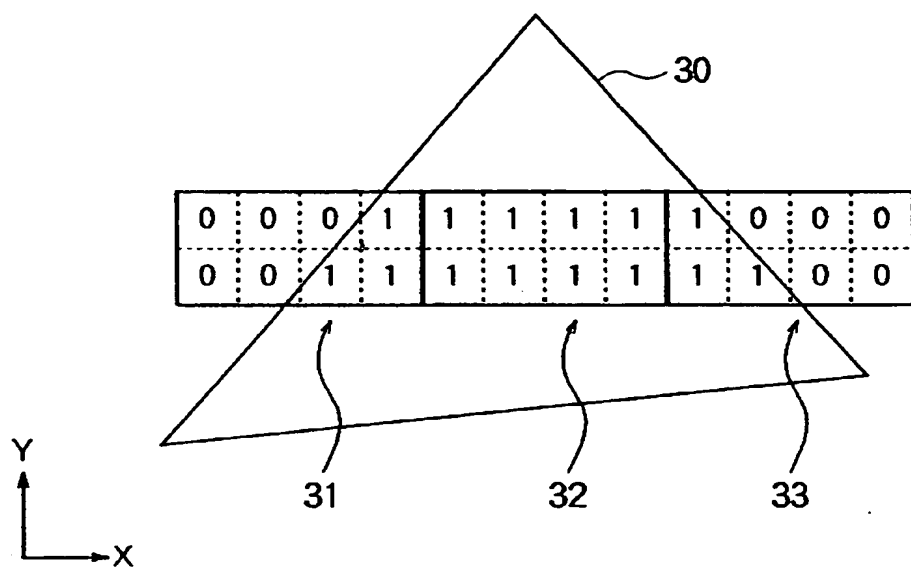




【図 1 1】



【図 1 2】



【書類名】 要約書

【要約】

【課題】 消費電力の大幅な低下を図れる画像処理装置を提供する

【解決手段】 ディスプレイに表示する所定の形状を単位図形の組み合わせで表現するために、複数の画素についての演算を同時に行い、処理対象となっている前記単位図形の内側に位置する画素についての演算結果を有効なものとして用いて処理を行なうグラフィック演算装置であって、演算サブブロック $200_1 \sim 205_8$ において、それぞれ対応するvalデータ $S220_1 \sim S220_8$ の有効性がクロックイネーブラ $210_1 \sim 215_8$ で判断され、対応するvalデータが有効を示す演算サブブロックのみが演算を実行し、そうでない演算サブブロックは演算を行なわない。演算ブロック $200 \sim 205$ は、パイプライン処理を実現する。

【選択図】 図3

認定・付加情報

特許出願の番号	平成11年 特許願 第051795号
受付番号	59900178199
書類名	特許願
担当官	第八担当上席 0097
作成日	平成11年 3月11日

<認定情報・付加情報>

【特許出願人】

【識別番号】

000002185

【住所又は居所】

東京都品川区北品川6丁目7番35号

【氏名又は名称】

ソニー株式会社

【代理人】

申請人

【識別番号】

100094053

【住所又は居所】

東京都台東区柳橋2丁目4番2号 創進国際特許  
事務所

【氏名又は名称】

佐藤 隆久

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日	1990年 8月30日
[変更理由]	新規登録
住 所	東京都品川区北品川6丁目7番35号
氏 名	ソニー株式会社